

COMPUTABLE RANDOMNESS AND BETTING FOR COMPUTABLE PROBABILITY SPACES

JASON RUTE

ABSTRACT. Unlike Martin-Löf randomness and Schnorr randomness, computable randomness has not been defined, except for a few ad hoc cases, outside of Cantor space. This paper offers such a definition (actually, many equivalent definitions), and further, provides a general method for abstracting “bit-wise” definitions of randomness from Cantor space to arbitrary computable probability spaces. This same method is also applied to give machine characterizations of computable and Schnorr randomness for computable probability spaces, extending the previous known results. This paper also addresses “Schnorr’s Critique” that gambling characterizations of Martin-Löf randomness are not computable enough. The paper contains a new type of randomness—endomorphism randomness—which the author hopes will shed light on the open question of whether Kolmogorov-Loveland randomness is equivalent to Martin-Löf randomness. It ends with ideas on how to extend this work to layerwise-computable structures, non-computable probability spaces, computable topological spaces, and measures defined by π -systems. It also ends with a possible definition of K-triviality for computable probability spaces.

1. INTRODUCTION

The subjects of measure theory and probability are filled with a number of theorems stating that some property holds “almost everywhere” or “almost surely.” Informally, these theorems state that if one starts with a random point, then the desired result is true. The field of algorithmic randomness has been very successful in making this notion formal: by restricting oneself to computable tests for non-randomness, one can achieve a measure-one set of points that behave as desired. The most prominent such notion of randomness is Martin-Löf randomness. However, Schnorr [35] gave an argument—which is now known as Schnorr’s Critique—that Martin-Löf randomness does not have a sufficiently computable characterization. He offered two weaker-but-more-computable alternatives: Schnorr randomness and computable randomness. All three randomness notions are interesting and robust, and further each has been closely linked to computable analysis (for example [12, 19, 33, 38]).

Computable randomness, however, is the only one of the three that has not been defined for arbitrary computable probability spaces. The usual definition is specifically for Cantor space (i.e. the space 2^ω of infinite binary strings), or by analogy, spaces such as 3^ω . Namely, a string $x \in 2^\omega$ is said to be computably random (in the fair-coin measure) if, roughly speaking, one cannot win arbitrarily large amounts of money using a computable betting strategy to gamble on the bits

2010 *Mathematics Subject Classification.* 03D32, 68Q30.

Submitted version. **Feedback and comments are still appreciated!**

This work has been partially supported by NSF grant DMS1068829.

of x . (See Section 2 for a formal definition.) While it is customary to say a real $x \in [0, 1]$ is computably random if its binary expansion is computably random in 2^ω , it was only recently shown [12] that this is the same as saying that, for example, the ternary expansion of x is computably random in 3^ω . In other words, computable randomness is base invariant.

In this paper, I use a method for extending the “bit-wise” definitions of randomness on Cantor space to arbitrary computable probability spaces. The method is based on previous methods given by Gács [18] and later Hoyrup and Rojas [23] of dividing a space into cells. However, to successfully extend a randomness notion (such that the new definition agrees with the former on 2^ω), one must show a property similar to base invariance. I do this for computable randomness.

An outline of the paper is as follows. Section 2 defines computable randomness on 2^ω , both for the fair-coin measure and for other computable probability measures on Cantor space. Unlike previous treatments (for example Bienvenu and Merkle [6]) I address the important pathological case where the measure may have null open sets.

Section 3 gives background on computable analysis, computable probability spaces, and algorithmic randomness.

Section 4 presents the concepts of an almost-everywhere decidable set (due to Hoyrup and Rojas [23]) and an a.e. decidable cell decomposition (which is similar to work of Hoyrup and Rojas [23] and Gács [18]). Recall that the topology of 2^ω is generated by the collection of basic open sets of the form $[\sigma]^\prec = \{x \in 2^\omega \mid x \succ \sigma\}$ where $x \succ \sigma$ means σ is an initial segment of x . Further, any Borel measure μ of 2^ω is determined by the values $\mu([\sigma]^\prec)$. The main idea of this paper is that for a computable probability space (\mathcal{X}, μ) one can replace the basic open sets of 2^ω (which are decidable) with an indexed family of “almost-everywhere decidable” sets $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$ which behave in much the same way. I call each such indexed family a cell decomposition of the space. This allows one to effortlessly transfer a definition from Cantor space to any computable probability space.

Section 5 applies this method to computable randomness, giving a variety of equivalent definitions based on martingales and other tests. More importantly, I show this definition is invariant under the choice of cell decomposition. Similar to the base-invariance proof of Brattka, Miller and Nies [12, 37], my proof uses computable analysis. However, their method does not apply here. (Their proof uses differentiability and the fact that every atomless measure on $[0, 1]$ is naturally equivalent to a measure on 2^ω . The situation is more complicated in the general case. One does not have differentiability, and one must consider absolutely-continuous measures instead of mere atomless ones.)

Section 6 gives a machine characterization of computable and Schnorr randomness for computable probability spaces. This combines the machine characterizations of computable randomness and Schnorr randomness (respectively, Mihailović [14, Theorem 7.1.25] and Downey, Griffiths, and LaForte [13]) with the machine characterization of Martin-Löf randomness on arbitrary computable probability spaces (Gács [17] and Hoyrup and Rojas [23]).

Section 7 shows a correspondence between cell decompositions of a computable probability space (\mathcal{X}, μ) and isomorphisms from (\mathcal{X}, μ) to Cantor space. I also show

computable randomness is preserved by isomorphisms between computable probability spaces, giving yet another characterization of computable randomness. However, unlike other notions of randomness, computable randomness is not preserved by mere morphisms (almost-everywhere computable, measure-preserving maps).

Section 8 gives three equivalent methods to extend a randomness notion to all computable probability measures. It also gives the conditions for when this new randomness notion agrees with the original one.

Section 9 asks how the method of this paper applies to Kolmogorov-Loveland randomness, another notion of randomness defined by gambling. The result is that the natural extension of Kolmogorov-Loveland randomness to arbitrary computable probability measures is Martin-Löf randomness. However, I do not answer the important open question as to whether Kolmogorov-Loveland randomness and Martin-Löf randomness are equivalent. Nonetheless, I do believe this answers Schnorr's Critique, namely that Martin-Löf randomness does have a natural definition in terms of computable betting strategies.

Section 10 explores a new notion of randomness in between Martin-Löf randomness and Kolmogorov-Loveland randomness, possibly equal to both. It is called endomorphism randomness.

Last, in Section 11, I suggest ways to generalize the method of this paper to a larger class of isomorphisms and cell decompositions. I also suggest methods for extending computable randomness to a larger class of probability spaces, including non-computable probability spaces, computable topological spaces, and measures defined by π -systems. Drawing on Section 6, I suggest a possible definition of K-triviality for computable probability spaces. Finally, I ask what can be known about the interplay between randomness, morphisms, and isomorphisms.

2. COMPUTABLE RANDOMNESS ON 2^ω

Before exploring computable randomness on arbitrary computable probability spaces, a useful intermediate step will be to consider computable probability measures on Cantor space.

We fix notation: $2^{<\omega}$ is the space of finite binary strings; 2^ω is the space of infinite binary strings; ε is the empty string; $\sigma \prec \tau$ and $\sigma \prec x$ mean σ is a proper initial segment of $\tau \in 2^{<\omega}$ or $x \in 2^\omega$; and $[\sigma]^\prec = \{x \in 2^\omega \mid \sigma \prec x\}$ is a BASIC OPEN SET or CYLINDER SET. Also for $\sigma \in 2^{<\omega}$ (or $x \in 2^\omega$), $\sigma(n)$ is the n th digit of σ (where $\sigma(0)$ is the "0th" digit) and $\sigma \upharpoonright n = \sigma(0) \cdots \sigma(n-1)$.

Typically, a MARTINGALE (on the fair-coin probability measure) is defined as a function $M: 2^{<\omega} \rightarrow [0, \infty)$ such that the following property holds for each $\sigma \in 2^{<\omega}$: $M(\sigma) = \frac{1}{2}(M(\sigma 0) + M(\sigma 1))$. Such a martingale can be thought of as a betting strategy on coins flips: the gambler starts with the value $M(\varepsilon)$ as her capital (where ε is the empty string) and bets on fair coin flips. Assuming the string σ represents the sequence of coin flips she has seen so far, $M(\sigma 0)$ is the resulting capital she has if the next flip comes up tails, and $M(\sigma 1)$ if heads. A martingale M is said to be COMPUTABLE if the value $M(\sigma)$ is uniformly computable from each σ .

A martingale M is said to SUCCEED on a string $x \in 2^\omega$ if $\limsup_{n \rightarrow \infty} M(x \upharpoonright n) = \infty$ (where $x \upharpoonright n$ is the first n bits of x), i.e. the gambler wins arbitrary large amounts of money using the martingale M while betting on the sequence x of flips. By Kolmogorov's theorem (see [14, Theorem 6.3.3]), such a martingale can only succeed on a measure-zero set of points. A string $x \in 2^\omega$ is said to be COMPUTABLY

RANDOM (on the fair-coin probability measure) if there does not exist a computable martingale M which succeeds on x .

Definition 2.1. A finite Borel measure μ on 2^ω is a COMPUTABLE MEASURE if the measure $\mu([\sigma]^\prec)$ of each basic open set is computable from σ . Further, if $\mu(2^\omega) = 1$, then we say μ is a COMPUTABLE PROBABILITY MEASURE (on 2^ω) and $(2^\omega, \mu)$ is a COMPUTABLE PROBABILITY SPACE (on 2^ω).

In this paper, measure always means a finite Borel measure. When convenient, I will drop the brackets and write $\mu(\sigma)$ instead. By the Carathéodory extension theorem, one may uniquely represent a computable measure as a computable function $\mu: 2^{<\omega} \rightarrow [0, \infty)$ such that

$$\mu(\sigma 0) + \mu(\sigma 1) = \mu(\sigma)$$

for all $\sigma \in 2^{<\omega}$. I will use often confuse a computable measure on 2^ω with its representation on $2^{<\omega}$.

The FAIR-COIN PROBABILITY MEASURE (or the LEBESGUE MEASURE on 2^ω) is the measure λ on 2^ω , defined by

$$\lambda(\sigma) = 2^{-|\sigma|}$$

where $|\sigma|$ is the length of σ . (The Greek letter λ will always be the fair-coin measure on 2^ω , except in a few examples where it is the Lebesgue measure on $[0, 1]^d$ or the uniform measure on 3^ω .)

One may easily generalize the definitions of martingale and computable randomness to a computable probability measure μ . The key idea is that the fairness condition still holds, but is now “weighted” by μ .

Definition 2.2. If μ is a computable probability measure on 2^ω , then a MARTINGALE M (with respect to the measure μ) is a partial function $M: 2^{<\omega} \rightarrow [0, \infty)$ such that the following two conditions hold:

- (1) (Fairness condition) For all $\sigma \in 2^{<\omega}$

$$M(\sigma 0)\mu(\sigma 0) + M(\sigma 1)\mu(\sigma 1) = M(\sigma)\mu(\sigma).$$

- (2) (Impossibility condition) $M(\sigma)$ is defined if and only if $\mu(\sigma) > 0$.

We say M is a COMPUTABLE MARTINGALE if $M(\sigma)$ is uniformly computable from σ (assuming $\mu(\sigma) > 0$).

Definition 2.3. Given a computable probability space $(2^\omega, \mu)$, a martingale M on $(2^\omega, \mu)$ and $x \in 2^\omega$, we say M SUCCEEDS on x if and only if $\limsup_{n \rightarrow \infty} M(x \upharpoonright n) = \infty$. Further, given $x \in 2^\omega$, if x is not in any measure-zero basic open set and there does not exist a computable martingale M on $(2^\omega, \mu)$ which succeeds on x , then we say x is COMPUTABLY RANDOM with respect to the measure μ .

Remark 2.4. The above definitions have been given before by Bienvenu and Merkle [6], and Definition 2.2 is an instance of the more general concept of martingale in probability theory (see for example Williams [40]).

The impossibility condition of Definition 2.2 follows from the slogan in probability theory that a measure-zero (or impossible) event can be ignored. A measure μ such that every open set has measure greater than zero is called a STRICTLY-POSITIVE measure. (Bienvenu and Merkle use the term “nowhere vanishing.”) Hence, the impossibility condition is not necessary when μ is strictly positive.

If $(2^\omega, \mu)$ is a strictly-positive probability space, then it is an easy folklore result that there is a bijection between computable martingales M and computable measures ν given by

$$\nu(\sigma) = M(\sigma)\mu(\sigma) \quad \text{and} \quad M(\sigma) = \nu(\sigma)/\mu(\sigma).$$

Even in the case where μ is not strictly positive, the impossibility condition guarantees that these equations can be used to define a computable measure from a computable martingale and vice-versa (under the conditions that $\text{undefined} \cdot 0 = 0$ and $x/0 = \text{undefined}$ for all x). Further, ν is always computable from M . Indeed, compute $\nu(\sigma)$ by recursion on the length of σ as follows. Since $\mu(\varepsilon) = 1$, $\nu(\varepsilon)$ is computable. To compute, say, $\nu(\sigma 0)$ from $\nu(\sigma)$, use

$$\nu(\sigma 0) = \begin{cases} M(\sigma 0)\mu(\sigma 0) & \text{if } \mu(\sigma 0) > 0 \\ \nu(\sigma) - M(\sigma 1)\mu(\sigma 1) & \text{if } \mu(\sigma 1) > 0 \\ 0 & \text{otherwise} \end{cases}.$$

This is computable, since in the case that $\mu(\sigma) = \mu(\sigma 0) = \mu(\sigma 1) = 0$, the bounds $0 \leq \nu(\sigma 0) \leq \nu(\sigma)$ “squeeze” $\nu(\sigma 0)$ to 0. Conversely, M can be computed from ν by waiting until $\mu(\sigma) > 0$, else $M(\sigma)$ is never defined.

Remark 2.5. It is possible to eliminate the impossibility condition altogether by considering martingales defined on the extended real numbers, i.e. $M: 2^{<\omega} \rightarrow [0, \infty]$. (Use the usual measure-theoretic convention that $\infty \cdot 0 = 0$.) Consider the martingale M_0 defined by $M_0(\sigma) = \lambda(\sigma)/\mu(\sigma)$ where λ is the fair-coin measure. Since, $\lambda(\sigma) > 0$ for all σ , we have that M_0 is computable on the extended real numbers. Notice $M_0(\sigma) = \infty$ if and only if $\mu(\sigma) = 0$, hence one can “forget” the infinite values to get a computable finite-valued martingale M_1 as in Definition 2.2. For any $x \in 2^\omega$, if M_0 succeeds on x then either $\mu(x \upharpoonright n) = 0$ for some n or $M_1(x)$ succeeds on x . In either case, x is not computably random. Conversely, if $x \in 2^\omega$ is not computably random, either M_0 succeeds on x or there is some finite-valued martingale M as in Definition 2 which succeeds on x . In the later case, $N = M + M_0$ is a martingale computable on the extended real numbers which also succeeds on x . However, this paper will only use the finite-valued martingales as in Definition 2.2.

I leave as an open question whether computable randomness can be defined on non-strictly positive measures without the impossibility condition and without infinite values.

Question 2.6. *Let μ be a computable probability measure on 2^ω , and assume x is not computably random on μ . Is there necessary a computable martingale $M: 2^{<\omega} \rightarrow [0, \infty)$ with respect to μ which is total, finite-valued and succeeds on x ?*

See Downey and Hirschfeld [14, Section 7.1] and Nies [32, Chapter 7] for more information on computable randomness for $(2^\omega, \lambda)$.

3. COMPUTABLE PROBABILITY SPACES AND ALGORITHMIC RANDOMNESS

In this section I give some background on computable analysis, computable probability spaces, and algorithmic randomness.

3.1. Computable analysis and computable probability spaces. Here I present computable Polish spaces and computable probability spaces. For a more detailed exposition of the same material see Hoyrup and Rojas [23]. This paper assumes some familiarity with basic computability theory and computable analysis, as in Pour El and Richards [34], Weihrauch [39], or Brattka et al. [11].

Definition 3.1. A COMPUTABLE POLISH SPACE (or COMPUTABLE METRIC SPACE) is a triple (X, d, S) such that

- (1) X is a complete metric space with metric $d: X \times X \rightarrow [0, \infty)$.
- (2) $S = \{a_i\}_{i \in \mathbb{N}}$ is a countable dense subset of X (the SIMPLE POINTS of \mathcal{X}).
- (3) The distance $d(a_i, a_j)$ is computable uniformly from i and j .

A point $x \in X$ is said to be COMPUTABLE if there is a computable CAUCHY-NAME $h \in \mathbb{N}^\omega$ for x , i.e. h is a computable sequence of natural numbers such that $d(a_{h(k)}, x) \leq 2^{-k}$ for all k .

The BASIC OPEN BALLS are sets of the form $B(a, r) = \{x \in X \mid d(x, a) < r\}$ where $a \in S$ and $r > 0$ is rational. The Σ_1^0 sets (EFFECTIVELY OPEN SETS) are computable unions of basic open balls; Π_1^0 sets (EFFECTIVELY CLOSED SETS) are the complements of Σ_1^0 sets; Σ_2^0 sets are computable unions of Π_1^0 sets; and Π_2^0 sets are computable intersections of Σ_1^0 sets. A function $f: \mathcal{X} \rightarrow \mathbb{R}$ is COMPUTABLE (-LY CONTINUOUS) if for each Σ_1^0 set U in \mathbb{R} , the set $f^{-1}(U)$ is Σ_1^0 in \mathcal{X} (uniformly in U), or equivalently, there is an algorithm which sends every Cauchy-name of x to a Cauchy-name of $f(x)$. A function $f: \mathcal{X} \rightarrow [0, \infty]$ is LOWER SEMICOMPUTABLE if it is the supremum of a computable sequence of computable functions $f_n: \mathcal{X} \rightarrow [0, \infty)$.

A real x is said to be LOWER (UPPER) SEMICOMPUTABLE if $\{q \in \mathbb{Q} \mid q < x\}$ (respectively $\{q \in \mathbb{Q} \mid q > x\}$) is a c.e. set.

Definition 3.2. If $\mathcal{X} = (X, d, S)$ is a computable Polish space, then a Borel measure μ is a COMPUTABLE MEASURE on \mathcal{X} if the value $\mu(X)$ is computable, and for each Σ_1^0 set U , the value $\mu(U)$ is lower semicomputable uniformly from the code for U . A COMPUTABLE PROBABILITY SPACE is a pair (\mathcal{X}, μ) where \mathcal{X} is a computable Polish space, μ is a computable measure on \mathcal{X} , and $\mu(\mathcal{X}) = 1$.

While this definition of computable probability space may seem ad hoc, it turns out to be equivalent to a number of other definitions. In particular, the computable probability measures on \mathcal{X} are exactly the computable points in the space of probability measures under the Prokhorov metric. Also, a probability space is computable precisely if the integral operator is a computable operator on computable functions $f: \mathcal{X} \rightarrow [0, 1]$. See Hoyrup and Rojas [23] and Schröder [36] for details.

I will often confuse a metric space or a probability space with its set of points, e.g. writing $x \in \mathcal{X}$ or $x \in (\mathcal{X}, \mu)$ to mean that $x \in X$ where $\mathcal{X} = (X, d, S)$.

3.2. Algorithmic randomness. In this section I give background on algorithmic randomness. Namely, I present three types of tests for Martin-Löf and Schnorr randomness. In Section 5, I will generalize these tests to computable randomness, building off the work of Merkle, Mihailović and Slaman [28] (which is similar to that of Downey, Griffiths and LaForte [13]). I also present Kurtz randomness.

Throughout this section, let (\mathcal{X}, μ) be a computable probability space.

Definition 3.3. A MARTIN-LÖF TEST (with respect to (\mathcal{X}, μ)) is a computable sequence of Σ_1^0 sets (U_n) such that $\mu(U_n) \leq 2^{-n}$ for all n . A SCHNORR TEST is a

Martin-Löf test such that $\mu(U_n)$ is also uniformly computable from n . We say x is COVERED BY the test (U_n) if $x \in \bigcap_n U_n$.

Definition 3.4. We say $x \in \mathcal{X}$ is MARTIN-LÖF RANDOM (with respect to (\mathcal{X}, μ)) if there is no Martin-Löf test which covers x . We say x is SCHNORR RANDOM if there is no Schnorr test which covers x . We say x is KURTZ RANDOM (or WEAK RANDOM) if x is not in any null Π_1^0 set (or equivalently a null Σ_2^0 set).

It is easy to see that for all computable probability spaces

$$\text{Martin-Löf} \rightarrow \text{Schnorr} \rightarrow \text{Kurtz}$$

It is also well-known (see [14, 32]) on $(2^\omega, \lambda)$ that

$$(3.1) \quad \text{Martin-Löf} \rightarrow \text{Computable} \rightarrow \text{Schnorr} \rightarrow \text{Kurtz}$$

In the next section, after defining computable randomness for computable probability spaces, I will show (3.1) holds for all computable probability spaces.

In analysis it is common to adopt the slogan “anything that happens on a measure-zero set is negligible.” In this paper it will be useful to adopt the slogan “anything that happens on a measure-zero Σ_2^0 set is negligible,” or in other words, “we do not care about points that are not Kurtz random.” (The reason for this choice will become apparent and is due to the close relationship between Kurtz randomness and a.e. computability. Section 7 contains more discussion.)

Next, I mention two other useful tests.

Definition 3.5. A VITALI TEST (or SOLOVAY TEST) is a sequence of Σ_1^0 sets (U_n) such that $\sum_n \mu(U_n) < \infty$. We say x is VITALI COVERED by (U_n) if $x \in U_n$ for infinitely many n . An INTEGRAL TEST is a lower semicomputable function $g: \mathcal{X} \rightarrow [0, \infty]$ such that $\int g d\mu < \infty$.

Theorem 3.6. For $x \in \mathcal{X}$, the following are equivalent.

- (1) x is Martin-Löf random (respectively Schnorr random).
- (2) x is not Vitali covered by any Vitali test (respectively any Vitali test (U_n) such that $\sum_n \mu(U_n)$ is computable).
- (3) $g(x) < \infty$ for all integral tests g (respectively for all integral tests g such that $\int g d\mu$ is computable).

Remark 3.7. The term Vitali test was coined recently by Nies. For a history of the tests for Schnorr and Martin-Löf randomness see Downey and Hirschfeld [14]. The integral test characterization for Schnorr randomness is due to Miyabe [31] and was also independently communicated to me by Hoyrup and Rojas.

I will give Vitali and integral test characterizations of computable randomness in Section 5.

There are also martingale characterizations of Martin-Löf and Schnorr randomness for 2^ω , but they will not be needed.

4. ALMOST-EVERYWHERE DECIDABLE CELL DECOMPOSITIONS

The main thesis of this paper is that “bit-wise” definitions of randomness for 2^ω , such as computable randomness, can be extended to arbitrary computable probability spaces by replacing the basic open sets $[\sigma]^\prec$ on 2^ω with an indexed family $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$ of a.e. decidable sets. This is the thesis of Hoyrup and Rojas

[23]. My method is based off of theirs, although the presentation and definitions differ on a few key points.

Recall that a set $A \subseteq \mathcal{X}$ is DECIDABLE if both A and its complement $\mathcal{X} \setminus A$ are Σ_1^0 sets (equivalently A is both Σ_1^0 and Π_1^0). The intuitive idea is that from the code for any $x \in \mathcal{X}$, one may effectively decide if x is in A or its complement. On 2^ω , the cylinder sets $[\sigma]^\prec$ are decidable. Unfortunately, a space such as $\mathcal{X} = [0, 1]$ has no non-trivial clopen sets, and therefore no non-trivial decidable sets. However, using the idea that null measure sets can be ignored, we can use “almost-everywhere decidable sets” instead.

Definition 4.1 (Hoyrup and Rojas [23]). Let (\mathcal{X}, μ) be a computable probability space. A pair $U, V \subseteq \mathcal{X}$ is a μ -A.E. DECIDABLE PAIR if

- (1) U and V are both Σ_1^0 sets,
- (2) $U \cap V = \emptyset$, and
- (3) $\mu(U \cup V) = 1$.

A set A is a μ -A.E. DECIDABLE SET if there is a μ -a.e. decidable pair U, V such that $U \subseteq A \subseteq \mathcal{X} \setminus V$. The code for the μ -a.e. decidable set A is the pair of codes for the Σ_1^0 sets U and V .

Hoyrup and Rojas [23] also required that $U \cup V$ be dense for technical reasons. We will relax this condition, working under the principle that one can safely ignore null open sets. They also use the terminology “almost decidable set”.

Definition 4.1 is an effectivization of μ -CONTINUITY SET, i.e. a set with μ -null boundary. Notice, the set $\mathcal{X} \setminus (U \cup V)$ includes the topological boundary, but since we do not require $U \cup V$ to be dense, it may also include null open sets.

Not every Σ_1^0 set is a.e. decidable; for example, take a dense open set with measure less than one. However, any basic open ball $B(a, r)$ is a.e. decidable provided that $\{x \mid d(a, x) = r\}$ has null measure. (Again, if we require the boundary to be nowhere dense, the situation is more subtle. See the discussion in Hoyrup and Rojas [23].) Further, the closed ball $\overline{B}(a, r)$ is also a.e. decidable with the same code. Any two a.e. decidable sets with the same code will be considered the same for our purposes. Hence, I will occasionally say $x \in A$ (respectively $x \notin A$), when I mean $x \in U$ (respectively $x \notin V$) for the corresponding a.e. decidable pair (U, V) .

Also notice that if A and B are a.e. decidable, then the Boolean operations $\mathcal{X} \setminus A$, $A \cap B$ and $A \cup B$ are a.e. decidable with codes computable from the codes for A and B .

Definition 4.2 (Inspired by Hoyrup and Rojas [23]). Let (\mathcal{X}, μ) be a computable probability space, and let $\mathcal{A} = (A_i)$ be a uniformly computable sequence of a.e. decidable sets. Let \mathcal{B} be the closure of \mathcal{A} under finite Boolean combinations. We say \mathcal{A} is an (A.E. DECIDABLE) GENERATOR of (\mathcal{X}, μ) if given a Σ_1^0 set $U \subseteq \mathcal{X}$ one can find (effectively from the code of U) a c.e. family $\{B_j\}$ of sets in \mathcal{B} (where $\{B_j\}$ is possibly finite or empty) such that $U = \sum_j B_j$ a.e.

Notice each generator generates the Borel sigma-algebra of \mathcal{X} up to a μ -null set. Hoyrup and Rojas [23] show that not only does such a generator \mathcal{A} exist for each (\mathcal{X}, μ) , but it can be taken to be a basis of the topology, hence they call \mathcal{A} a “basis of almost decidable sets”. I will not require that \mathcal{A} is a basis.

Theorem 4.3 (Hoyrup and Rojas [23]). *Let (\mathcal{X}, μ) be a computable probability space. There exists an a.e. decidable generator \mathcal{A} of (\mathcal{X}, μ) . Further, \mathcal{A} is computable from (\mathcal{X}, μ) .*

The main idea of the proof for Theorem 4.3 is to start with the collection of basic open balls centered at simple points with rational radii. While, these may not have null boundary, a basic diagonalization argument (similar to the proof of the Baire category theory, see [10]) can be used to calculate a set of radii approaching zero for each simple point such that the resulting ball is a.e. decidable. Similar arguments have been given by Bosserhoff [9] and Gács [18]. The technique is related to Bishop's theory of profiles [8, Section 6.4] and to “derandomization” arguments (see Freer and Roy [16] for example).

From a generator we can decompose \mathcal{X} into a.e. decidable cells. This is the indexed family $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$ mentioned in the introduction.

Definition 4.4. Let $\mathcal{A} = (A_i)$ be an a.e. decidable generator of (\mathcal{X}, μ) . Recall each A_i is coded by an a.e. decidable pair (U_i, V_i) where $U_i \subseteq A_i \subseteq \mathcal{X} \setminus V_i$. For $\sigma \in 2^\omega$ of length s define $[\sigma]_{\mathcal{A}} = A_0^{\sigma(0)} \cap A_1^{\sigma(1)} \cap \dots \cap A_{s-1}^{\sigma(s-1)}$ where for each i , $A_i^0 = U_i$ and $A_i^1 = V_i$. When possible, define $x \upharpoonright_{\mathcal{A}} n$ as the unique σ of length n such that $x \in [\sigma]_{\mathcal{A}}$. Also when possible, define the \mathcal{A} -NAME of x as the string $\text{name}_{\mathcal{A}}(x) = \lim_{n \rightarrow \infty} x \upharpoonright_{\mathcal{A}} n$. A point without an \mathcal{A} -name will be called an UNREPRESENTED POINT. Each $[\sigma]_{\mathcal{A}}$ will be called a CELL, and the collection of $\{[\sigma]_{\mathcal{A}}\}_{\sigma \in 2^{<\omega}}$ will be called an (A.E. DECIDABLE) CELL DECOMPOSITION of (\mathcal{X}, μ) .

The choice of notation allows one to quickly translate between Cantor space and the space (\mathcal{X}, μ) . Gács [18] and others refer to the cell $[x \upharpoonright_{\mathcal{A}} n]_{\mathcal{A}}$ as the n -CELL of x and writes it as $\Gamma_n(x)$.

Remark 4.5. There are two types of “bad points”, unrepresented points and points $x \in [\sigma]_{\mathcal{A}}$ where $\mu([\sigma]_{\mathcal{A}}) = 0$. The set of “bad points” is a null Σ_2^0 set, so each “bad point” is not even Kurtz random! One may also go further, and for each generator \mathcal{A} compute another \mathcal{A}' such that $[\sigma]_{\mathcal{A}} = [\sigma]_{\mathcal{A}'}$ a.e., but $\mu([\sigma]_{\mathcal{A}}) = 0$ if and only if $[\sigma]_{\mathcal{A}'} = \emptyset$. Then all the “bad points” would be unrepresented points.

Example 4.6. Consider a computable measure μ on 2^ω . Let $A_i = \{x \in 2^\omega \mid x(i) = 1\}$ where $x(i)$ is the i th bit of x . Then $\mathcal{A} = (A_i)$ is a generator of $(2^\omega, \mu)$. Further $[\sigma]_{\mathcal{A}} = [\sigma]^\prec$, $x \upharpoonright_{\mathcal{A}} n = x \upharpoonright n$, and $\text{name}_{\mathcal{A}}(x) = x$. Call \mathcal{A} the NATURAL GENERATOR of $(2^\omega, \mu)$, and $\{[\sigma]^\prec\}_{\sigma \in 2^{<\omega}}$ the NATURAL CELL DECOMPOSITION.

In this next proposition, recall that a set $S \subseteq 2^{<\omega}$ is PREFIX-FREE if there is no pair $\tau, \sigma \in S$ such that $\tau \prec \sigma$.

Proposition 4.7. *Let (\mathcal{X}, μ) be a computable probability space with generator \mathcal{A} and $\{[\sigma]_{\mathcal{A}}\}_{\sigma \in 2^{<\omega}}$ the corresponding cell decomposition. Then for each Σ_1^0 set $U \subseteq \mathcal{X}$ there is a c.e. set $\{\sigma_i\}$ (computable from U) such that $U = \bigcup_i [\sigma_i]_{\mathcal{A}}$ a.e. Further, $\{\sigma_i\}$ can be assumed to be prefix-free and such that $\mu([\sigma_i]_{\mathcal{A}}) > 0$ for all i .*

Proof. Straight-forward from Definition 4.1. □

It is clear that a generator \mathcal{A} is computable from its cell decomposition $\{[\sigma]_{\mathcal{A}}\}_{\sigma \in 2^{<\omega}}$, namely let

$$A_i = \bigcup_{\{\sigma : \sigma(i)=1\}} [\sigma]_{\mathcal{A}}.$$

Hence we will often confuse a generator and its cell decomposition writing both as \mathcal{A} . Further, this next proposition gives the criterion for when an indexed family $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$ forms an a.e. decidable cell decomposition.

Proposition 4.8. *Let (\mathcal{X}, μ) be a computable probability space. Let $\mathcal{A} = \{A_\sigma\}_{\sigma \in 2^{<\omega}}$ be a computably indexed family of Σ_1^0 sets such that*

- (1) *for all $\sigma \in 2^\omega$, $A_{\sigma 0} \cap A_{\sigma 1} = \emptyset$ and $A_{\sigma 0} \cup A_{\sigma 1} = A_\sigma$ a.e.*
- (2) *$\mu(A_\varepsilon) = 1$, and*
- (3) *for each Σ_1^0 set $U \subseteq \mathcal{X}$ there is a c.e. set $\{\sigma_i\}$ (computable from U) such that $U = \bigcup_i [\sigma_i]_{\mathcal{A}}$ a.e.*

Then \mathcal{A} is an a.e. decidable cell decomposition where $[\sigma]_{\mathcal{A}} = A_\sigma$ a.e. for all $\sigma \in 2^{<\omega}$.

Proof. Straight-forward from Definition 4.1 and Definition 4.4. \square

Each computable probability space (\mathcal{X}, μ) is uniquely represented by a cell decomposition \mathcal{A} and the values $\mu([\sigma]_{\mathcal{A}})$.

The main difference between the method here and that of Gács [18] and Hoyrup and Rojas [23] is that they pick a canonical cell decomposition for each (\mathcal{X}, μ) . Also they assume every point $x \in \mathcal{X}$ is in some cell, and that no two points have the same \mathcal{A} -name. I, instead, work with all cell decompositions simultaneously and do not require as strong of properties. This will allow me to give a correspondence between cell decompositions and isomorphisms in Section 7.

5. COMPUTABLE RANDOMNESS ON COMPUTABLE PROBABILITY SPACES

In this section I define computable randomness on a computable probability space. As a first step, I have already done this for spaces $(2^\omega, \mu)$. The second step will be to define computable randomness with respect to a particular cell decomposition of the space. Finally, the last step is Theorem 5.6, where I will show the definition is invariant under the choice of cell decomposition.

There are two characterizations of computable randomness on $(2^\omega, \lambda)$ that use Martin-Löf tests. The first was due to Downey, Griffiths, and LaForte [13]. However, I will use another due to Merkle, Mihailović, and Slaman [28].

Definition 5.1 (Merkle et al. [28]). On $(2^\omega, \lambda)$ a Martin-Löf test (U_n) is called a **BOUNDED MARTIN-LÖF TEST** if there is a computable measure $\nu: 2^{<\omega} \rightarrow [0, \infty)$ such that for all $n \in \mathbb{N}$ and $\sigma \in 2^{<\omega}$

$$\mu(U_n \cap [\sigma]^\prec) \leq 2^{-n} \nu(\sigma).$$

We say that the test (U_n) is **BOUNDED BY** the measure ν .

Theorem 5.2 (Merkle et al. [28]). *On $(2^\omega, \lambda)$, a string $x \in 2^\omega$ is computably random if and only if x is not covered by any bounded Martin-Löf test.*

The next theorem and definition give five equivalent tests for computable randomness (with respect to a cell decomposition \mathcal{A}). (I also give a machine characterization of computable randomness in Section 6.) The integral test and Vitali cover test are new for computable randomness, although they are implicit in the proof of Theorem 5.2.

Theorem 5.3. *Let \mathcal{A} be a cell decomposition of the computable probability space (\mathcal{X}, μ) . If $x \in \mathcal{X}$ is neither an unrepresented point nor in a null cell, then the following are equivalent.*

- (1) (*Martingale test*) There is a computable martingale $M: 2^{<\omega} \rightarrow [0, \infty)$ satisfying

$$M(\sigma 0)\mu([\sigma 0]_{\mathcal{A}}) + M(\sigma 1)\mu([\sigma 1]_{\mathcal{A}}) = M(\sigma)\mu([\sigma]_{\mathcal{A}})$$

$$M(\sigma) \text{ is defined} \iff \mu([\sigma]_{\mathcal{A}}) > 0$$

for all $\sigma \in 2^{<\omega}$ such that $\limsup_{n \rightarrow \infty} M(x \upharpoonright_{\mathcal{A}} n) = \infty$.

- (2) (*Martingale test with savings property*, see for example [14, Proposition 2.3.8]) There is a computable martingale $N: 2^{<\omega} \rightarrow [0, \infty)$ satisfying the conditions of (1) and a partial-computable “savings function” $f: 2^{<\omega} \rightarrow [0, \infty)$ satisfying

$$f(\sigma) \leq N(\sigma) \leq f(\sigma) + 1$$

$$\sigma \preceq \tau \implies f(\sigma) \leq f(\tau)$$

$$f(\sigma) \text{ is defined} \iff \mu([\sigma]_{\mathcal{A}}) > 0$$

for all $\sigma, \tau \in 2^{<\omega}$ such that $\lim_{n \rightarrow \infty} N(x \upharpoonright_{\mathcal{A}} n) = \infty$.

- (3) (*Integral test*) There is a computable measure $\nu: 2^{<\omega} \rightarrow [0, \infty)$ and a lower semicomputable function $g: \mathcal{X} \rightarrow [0, \infty]$ satisfying

$$\int_{[\sigma]_{\mathcal{A}}} g d\mu \leq \nu(\sigma)$$

for all $\sigma \in 2^{<\omega}$ such that $g(x) = \infty$.

- (4) (*Bounded Martin-Löf test*) There is a computable measure $\nu: 2^{<\omega} \rightarrow [0, \infty)$ and a Martin-Löf test (U_n) satisfying

$$\mu(U_n \cap [\sigma]_{\mathcal{A}}) \leq 2^{-n} \nu(\sigma).$$

for all $n \in \mathbb{N}$ and $\sigma \in 2^{<\omega}$ such that (U_n) covers x .

- (5) (*Vitali cover test*) There is a computable measure $\nu: 2^{<\omega} \rightarrow [0, \infty)$ and a Vitali cover (V_n) satisfying

$$\sum_n \mu(V_n \cap [\sigma]_{\mathcal{A}}) \leq \nu(\sigma)$$

for all $n \in \mathbb{N}$ and $\sigma \in 2^{<\omega}$ such that (V_n) Vitali covers x .

For (3) through (5), the measure ν may be assumed to be a probability measure and satisfy the following absolute-continuity condition,

$$(5.1) \quad \nu(\sigma) \leq \int_{[\sigma]_{\mathcal{A}}} h d\mu$$

for some integrable function h .

Further, each test is uniformly computable from any other.

Definition 5.4. Let \mathcal{A} be a cell decomposition of the space (\mathcal{X}, μ) . Say $x \in X$ is **COMPUTABLY RANDOM** (with respect to \mathcal{A}) if x is neither an unrepresented point nor in a null cell, and x does not satisfy any of the equivalent conditions (1–5) of Theorem 5.3.

Proof of Theorem 5.3. (1) implies (2): The idea is to bet with the martingale M as usual, except at each stage set some of the winnings aside into a savings account $f(\sigma)$ and bet only with the remaining capital. Formally, define N and f recursively

as follows. (One may assume $M(\sigma) > 0$ for all σ by adding 1 to $M(\sigma)$.) Start with $N(\varepsilon) = M(\varepsilon)$ and $f(\varepsilon) = 0$. At σ , for $i = 0, 1$ let

$$N(\sigma i) = f(\sigma) + \frac{M(\sigma i)}{M(\sigma)}(N(\sigma) - f(\sigma))$$

and $f(\sigma i) = \max(f(\sigma), N(\sigma i) - 1)$.

(2) implies (3): Let $\nu(\sigma) = N(\sigma)\mu([\sigma]_{\mathcal{A}})$ and $g(x) = \sup_{s \rightarrow \infty} f(x \upharpoonright_{\mathcal{A}} s)$. Then $\int_{[\sigma]_{\mathcal{A}}} g \, d\mu \leq \nu(\sigma) \leq \int_{[\sigma]_{\mathcal{A}}} (g + 1) \, d\mu$, which also shows ν satisfies the absolute-continuity condition of formula (5.1). If $N(\varepsilon)$ is scaled to be 1, then ν is a probability measure.

(3) implies (1): Let $M(\sigma) = \nu(\sigma)/\mu([\sigma]_{\mathcal{A}})$. Then $M(x \upharpoonright_{\mathcal{A}} k) \geq \frac{\int_{[x \upharpoonright_{\mathcal{A}} k]_{\mathcal{A}}} g \, d\mu}{\mu([x \upharpoonright_{\mathcal{A}} k]_{\mathcal{A}})}$, which converges to ∞ .

(3) implies (4): Let $U_n = \{x \mid g(x) > 2^n\}$. By Markov's inequality, $\mu(U_n \cap [\sigma]_{\mathcal{A}}) \leq \int_{[\sigma]_{\mathcal{A}}} g \, d\mu \leq \nu(\sigma)$.

(4) implies (5): Let $V_n = U_n$.

(5) implies (3): Let $g = \sum_n \mathbf{1}_{V_n}$. □

In this next proposition, I show the standard randomness implications (as in formula (3.1)) still hold.

Proposition 5.5. *Let (\mathcal{X}, μ) be a computable probability space. If $x \in \mathcal{X}$ is Martin-Löf random, then x is computably random (with respect to every cell decomposition \mathcal{A}). If $x \in \mathcal{X}$ is computably random (with respect to a cell decomposition \mathcal{A}), then x is Schnorr random, and hence Kurtz random.*

Proof. The statement on Martin-Löf randomness follows from the bounded Martin-Löf test (Theorem 5.3 (4)).

For the last statement, assume x is not Schnorr random. If x is an unrepresented point or in a null cell, then x is not computably random by Definition 5.4. Else, there is some Vitali cover (V_n) where $\sum_n \mu(V_n)$ is computable and (V_n) Vitali-covers x . Define $\nu: 2^{<\omega} \rightarrow [0, \infty)$ as $\nu(\sigma) = \sum_n \mu(V_n \cap [\sigma]_{\mathcal{A}})$. Then clearly, $\mu(V_n \cap [\sigma]_{\mathcal{A}}) \leq \nu(\sigma)$ for all n and σ . By the Vitali cover test (Theorem 5.3 (5)), it is enough to show that ν is a computable measure. It is straightforward to verify that $\nu(\sigma 0) + \nu(\sigma 1) = \nu(\sigma)$. As for the computability of ν ; notice $\nu(\sigma)$ is lower semicomputable for each σ since μ is a computable probability measure (see Definition 3.2). Then since $\nu(\varepsilon) = \sum_n \mu(V_n)$ is computable, ν is a computable measure. □

Theorem 5.6. *The definition for computable randomness does not depend on the choice of cell decomposition.*

Proof. Before giving the details, here is the main idea. It suffices to convert a test with respect to one cell decomposition \mathcal{A} to another test which covers the same points, but is with respect to a different cell decomposition \mathcal{B} . In order to do this, take a bounding measure ν with respect to \mathcal{A} (which is really a measure on 2^ω) and transfer it to an actual measure π on \mathcal{X} . Then transfer π back to a bounding measure κ with respect to \mathcal{B} . In order to guarantee that this will work, we will assume ν satisfies the absolute-continuity condition of formula 5.1, which ensures that π exists and is absolutely continuous with respect to μ .

Now I give the details. Assume $x \in X$ is not computably random with respect to the cell decomposition \mathcal{A} of the space. Let \mathcal{B} be another cell decomposition. If

x is an unrepresented point or in a null cell, then x is not a Kurtz random point of (\mathcal{X}, μ) , and by Proposition 5.5, x is not computably random with respect to \mathcal{B} .

So assume x is neither an unrepresented point nor in a null cell. By condition (4) of Theorem 5.3 there is some Martin-Löf test (U_n) bounded by a probability measure ν such that (U_n) covers x . Further, ν can be assumed to satisfy the absolute-continuity condition in formula (5.1).

Claim. There is a computable probability measure π on \mathcal{X} defined by $\pi([\sigma]_{\mathcal{A}}) = \nu(\sigma)$ which is absolutely continuous with respect to μ , i.e. every μ -null set is a π -null set.

Proof of claim. This is basically the Carathéodory extension theorem. The collection $\{[\sigma]_{\mathcal{A}}\}_{\sigma \in 2^{<\omega}}$ is essentially a semi-ring. A semi-ring contains \emptyset , is closed under intersections, and for each A, B in the semi-ring, there are pairwise disjoint sets C_1, \dots, C_n in the semi-ring such that $A \setminus B = C_1 \cup \dots \cup C_n$. To make this collection a semi-ring which generates the Borel sigma-algebra, add every μ -null set and every set which is μ -a.e. equal to $[\sigma]_{\mathcal{A}}$ for some σ .

Define $\pi([\sigma]_{\mathcal{A}}) = \nu(\sigma)$ and $\pi(\emptyset) = 0$ and similarly for the μ -a.e. equivalent sets. (This is well defined since if $[\sigma]_{\mathcal{A}} = [\tau]_{\mathcal{A}}$ μ -a.e. then by the absolute continuity condition, $\nu(\sigma) = \nu(\tau)$, and similarly if $\mu([\sigma]_{\mathcal{A}}) = 0$, then $\nu(\sigma) = 0$.) Now, it is enough to show π is a pre-measure, specifically that it satisfies countable additivity. Assume for some pairwise disjoint family $\{A_i\}$ and some B , both in the semi-ring, that $B = \bigcup_i A_i$. If B is μ -null, then each A_i is as well. By the definition of π on μ -null sets, we have $\pi(B) = 0 = \sum_i \pi(A_i)$. If B is not μ -null, then $B = [\tau]_{\mathcal{A}}$ μ -a.e. for some τ and each A_i of positive μ -measure is μ -a.e. equal to $[\sigma_i]_{\mathcal{A}}$ for some $\sigma_i \succeq \tau$. For each k , let $C_k = [\tau]_{\mathcal{A}} \setminus \bigcup_{i=0}^{k-1} [\sigma_i]_{\mathcal{A}}$, which is a finite union of basic open sets in 2^ω . Let D_k be the same union as C_k but replacing each $[\sigma]_{\mathcal{A}}$ with $[\sigma]_{\mathcal{B}}$. Then by the absolute continuity condition,

$$\pi(B) - \sum_{i=0}^{k-1} \pi(A_i) = \nu(\tau) - \sum_{i=0}^{k-1} \nu(\sigma_i) = \nu(C_k) = \int_{D_k} h d\mu$$

Since $[\tau]_{\mathcal{A}} = \bigcup_i [\sigma_i]_{\mathcal{A}}$ μ -a.e., the right-hand-side goes to zero as $k \rightarrow \infty$. So π is a pre-measure and may be extended to a measure by the Carathéodory extension theorem.

Similarly by approximation, π satisfies $\pi(A) \leq \int_A h d\mu$ for all Borel sets A and hence is absolutely continuous with respect to μ .

To see π is a computable probability measure on \mathcal{X} , take a Σ_1^0 set U . By Proposition 4.7, there is a c.e., prefix-free set $\{\sigma_i\}$ of finite strings such that $U = \bigcup_i [\sigma_i]_{\mathcal{A}}$ μ -a.e. (and so π -a.e. by absolute continuity). As this union is disjoint, $\pi(U) = \sum_i \pi([\sigma_i]_{\mathcal{A}}) = \sum_i \nu(\sigma_i)$ μ -a.e. and so $\pi(U)$ is lower-semicomputable. Since $\pi(\mathcal{X}) = 1$, π is a computable probability measure. This proves the claim. \square

Let π be as in the claim. Since π is absolutely continuous with respect to μ , any a.e. decidable set of μ is an a.e. decidable set of π . In particular, the measures $\pi([\tau]_{\mathcal{B}})$ are computable from τ . Now transfer π back to a measure $\kappa: 2^{<\omega} \rightarrow [0, \infty)$ using $\kappa(\sigma) = \pi([\sigma]_{\mathcal{B}})$. This is a computable probability measure since $\pi([\sigma]_{\mathcal{B}})$ is computable.

Last, we show the Martin-Löf test (U_n) is bounded by κ with respect to the cell decomposition \mathcal{B} . To see this, fix $\tau \in 2^{<\omega}$ and take the c.e., prefix-free set $\{\sigma_i\}$ of

finite strings such that $[\tau]_{\mathcal{B}} = \bigcup_i [\sigma_i]_{\mathcal{A}}$ μ -a.e. (and so π -a.e.). Then $\kappa(\tau) = \sum_i \nu(\sigma_i)$, and for each n ,

$$\mu(U_n \cap [\tau]_{\mathcal{B}}) = \sum_i \mu(U_n \cap [\sigma_i]_{\mathcal{A}}) \leq \sum_i 2^{-n} \nu(\sigma_i) = 2^{-n} \kappa(\tau). \quad \square$$

Theorem 5.3 is just a sample of the many equivalent definitions for computable randomness. I conjecture that the other known characterizations of computable randomness, see for example Downey and Hirschfeld [14, Section 7.1], can be extended to arbitrary computable Polish spaces using the techniques above. As well, other test characterizations for Martin-Löf randomness can be extended to computable randomness by “bounding the test” with a computable measure or martingale. (See Section 6 for an example using machines.) Further, the proof of Theorem 5.6 shows that the bounding measure ν can be assumed to be a measure on \mathcal{X} , instead of 2^ω , under the additional condition that \mathcal{A} is a cell decomposition for both (\mathcal{X}, μ) and (\mathcal{X}, ν) . Similarly, we could modify the martingale test to assume M is a martingale on (\mathcal{X}, μ) (in the sense of probability theory) with an appropriate filtration.

Actually, the above ideas can be used to show any L^1 -bounded a.e. computable martingale (in the sense of probability theory) converges on computable randoms if the filtration converges to the Borel sigma-algebra (or even a “computable” sigma-algebra) and the L^1 -bound is computable. This can be extended to (the Schnorr layerwise-computable representatives of) L^1 -computable martingales as well. The proof is beyond the scope of this paper and will be published separately.

In Section 11, I give ideas on how computable randomness can be defined on an even broader class of spaces, and also on non-computable probability spaces. I end this section by showing that Definition 5.4 is consistent with the usual definitions of computable randomness on 2^ω , Σ^ω , and $[0, 1]$.

Example 5.7. Consider a computable probability measure μ on 2^ω . It is easy to see that computable randomness in the sense of Definition 5.4 with respect to the natural cell decomposition is equivalent to computable randomness on 2^ω as defined in Definition 2.3. Since Definition 5.4 is invariant under the choice of cell decomposition (Theorem 5.6), the two definitions agree on $(2^\omega, \mu)$.

Example 5.8. Consider a computable probability measure μ on Σ^ω where $\Sigma = \{a_0, \dots, a_{k-1}\}$ is a finite alphabet. It is natural to define a martingale $M: \Sigma^\omega \rightarrow [0, \infty)$ as one satisfying the fairness condition

$$M(\sigma a_0)\mu(\sigma a_0) + \dots + M(\sigma a_{k-1})\mu(\sigma a_{k-1}) = M(\sigma)\mu(\sigma)$$

for all $\sigma \in \Sigma^{<\omega}$ (along with the impossibility condition from Definition 2.2). A little thought reveals that by systematically grouping and upgrouping cylinder sets M can be turned into a binary martingale which succeeds on the same points. For example, given a martingale M on 3^ω , one may first split $[\sigma]^\prec$ into $[\sigma 0]^\prec$ and $A_\sigma = [\sigma 1]^\prec \cup [\sigma 2]^\prec$. Define,

$$M(A_\sigma) = \frac{M(\sigma 1)\mu(\sigma 1) + M(\sigma 2)\mu(\sigma 2)}{\mu([\sigma 1]^\prec \cup [\sigma 2]^\prec)}$$

and notice the fairness condition is still satisfied,

$$M(\sigma 0)\mu(\sigma 0) + M(A_\sigma)\mu(A_\sigma) = M(\sigma)\mu(\sigma).$$

In the next step, one may split A_σ into $[\sigma 1]^\prec$ and $[\sigma 2]^\prec$ to give

$$M(\sigma 1)\mu(\sigma 1) + M(\sigma 2)\mu(\sigma 2) = M(A_\sigma)\mu(A_\sigma).$$

This grouping and ungrouping of cylinder sets forms a (binary) cell decomposition \mathcal{A} on $(3^\omega, \mu)$. If M was first given the savings property, this new martingale succeeds on the same points. It follows that $x \in 3^\omega$ is computably random in the natural sense if and only if it is computably random as in Definition 5.4.

Example 5.9. Let $([0, 1], \lambda)$ be the space $[0, 1]$ with the Lebesgue measure. Let $A_i = \{x \in [0, 1] \mid \text{the } i\text{th binary digit of } x \text{ is } 1\}$. Then $\mathcal{A} = (A_i)$ is a generator of $([0, 1], \lambda)$ and $[\sigma]_{\mathcal{A}} = [0.\sigma, 0.\sigma + 2^{-|\sigma|}]$ a.e. A little thought reveals that $x \in ([0, 1], \lambda)$ is computably random (in the sense of Definition 5.4) if and only if the binary expansion of x is computably random in $(2^\omega, \lambda)$ with the fair-coin measure. This is the standard definition of computable randomness on $([0, 1], \lambda)$. Further, using a base b other than binary gives a different generator, for example let $A_{bi+j} = \{x \in [0, 1] \mid \text{the } i\text{th } b\text{-ary digit of } x \text{ is } j\}$ where $0 \leq j < b$. Yet, the computably random points remain the same. Hence computable randomness on $([0, 1], \lambda)$ is base invariant [12, 37]. (The proof of Theorem 5.6 has similarities to the proof of Brattka, Miller and Nies [12], but as mentioned in the introduction, there are also key differences.) Also see Example 7.11.

More examples are given at the end of Section 7.

6. MACHINE CHARACTERIZATIONS OF COMPUTABLE AND SCHNORR RANDOMNESS

In this section I give machine characterizations of computable and Schnorr randomness for computable probability spaces. This has already been done for Martin-Löf randomness.

Recall the following definition and fact.

Definition 6.1. A machine M is a partial-computable function $M: 2^{<\omega} \rightarrow 2^{<\omega}$. A machine is PREFIX-FREE if $\text{dom } M$ is prefix-free. The prefix-free Kolmogorov complexity of σ relative to a machine M is

$$K_M(\sigma) = \inf \{|\tau| \mid \tau \in 2^{<\omega} \text{ and } M(\tau) = \sigma\}.$$

(There is a non-prefix-free version of complexity as well.)

Theorem 6.2 (Schnorr (see [14, Theorem 6.2.3])). *A string $x \in (2^\omega, \lambda)$ is Martin-Löf random if and only if for all prefix-free machines M ,*

$$(6.1) \quad \limsup_{n \rightarrow \infty} (n - K_M(x \upharpoonright n)) < \infty.$$

Schnorr's theorem has been extended to both Schnorr and computable randomness.

Definition 6.3. For a machine M define the semimeasure $\text{meas}_M: 2^{<\omega} \rightarrow [0, \infty)$ as

$$\text{meas}_M(\sigma) = \sum_{\substack{\tau \in \text{dom } M \\ M(\tau) \succeq \sigma}} 2^{-|\tau|}.$$

A machine M is a COMPUTABLE-MEASURE MACHINE if $\text{meas}_M(\varepsilon)$ is computable. A machine M is a BOUNDED MACHINE if there is some computable-measure ν such that $\text{meas}_M(\sigma) \leq \nu(\sigma)$ for all $\sigma \in 2^{<\omega}$.

Downey, Griffiths, and LaForte [13] showed that $x \in (2^\omega, \lambda)$ is Schnorr random precisely if formula (6.1) holds for all prefix-free, computable-measure machines. Mihailović (see [14, Theorem 7.1.25]) showed that $x \in (2^\omega, \lambda)$ is computably random precisely if formula (6.1) holds for all prefix-free, bounded machines.

Schnorr's theorem was extended to all computable probability measures on Cantor space by Gács [17]. Namely, replace formula (6.1) with

$$\limsup_{n \rightarrow \infty} (-\log_2 \mu([x \upharpoonright n]^\prec) - K_M(x \upharpoonright n)) < \infty.$$

If $\mu([x \upharpoonright n]) = 0$ for any n then we say this inequality is false. Hoyrup and Rojas [23] extended this to any computable probability space. Here, I do the same for Schnorr and computable randomness (I include Martin-Löf randomness for completeness).

Theorem 6.4. *Let (\mathcal{X}, μ) be a computable probability space and $x \in \mathcal{X}$.*

(1) *$x \in \mathcal{X}$ is Martin-Löf random precisely if*

$$(6.2) \quad \limsup_{n \rightarrow \infty} (-\log_2 \mu([x \upharpoonright_A n]_A) - K_M(x \upharpoonright_A n)) < \infty.$$

holds for all prefix-free machines M . (Again, we say formula (6.2) is false if $\mu([x \upharpoonright n]) = 0$ for any n .)

- (2) *$x \in \mathcal{X}$ is computably random precisely if formula (6.2) holds for all prefix-free, computable-measure machines M .*
- (3) *$x \in \mathcal{X}$ is Schnorr random precisely if formula (6.2) holds for all prefix-free, bounded machines M .*

Further, (1) through (3) hold even if M is not assumed to be prefix-free, but only that $\text{meas}_M(\varepsilon) \leq 1$.

Proof. Slightly modify the proofs of Theorems 6.2.3, 7.1.25, and 7.1.15 in Downey and Hirschfeldt [14], respectively. \square

7. COMPUTABLE RANDOMNESS AND ISOMORPHISMS

In this section I give another piece of evidence that the definition of computable randomness in this paper is robust, namely that the computably random points are preserved under isomorphisms between computable probability spaces. I also show a one-to-one correspondence between cell decompositions of a computable measure space and isomorphisms from that space to the Cantor space.

Definition 7.1. Let (\mathcal{X}, μ) and (\mathcal{Y}, ν) be computable probability spaces.

- (1) A partial map $T: \mathcal{X} \rightarrow \mathcal{Y}$ is said to be PARTIAL COMPUTABLE if there is a partial-computable function $F: \mathbb{N}^\omega \rightarrow \mathbb{N}^\omega$ which given a Cauchy-name for $x \in \mathcal{X}$ returns the Cauchy-name for $T(x)$, and further, the domain of T is maximal for this h , i.e. $x \in \text{dom}(T)$ if and only if for all $a, b \in \mathbb{N}^\omega$ which are Cauchy-names for x , then $a, b \in \text{dom}(F)$ and both $F(a)$ and $F(b)$ are Cauchy-names for the same point in \mathcal{Y} .
- (2) A partial map $T: (\mathcal{X}, \mu) \rightarrow \mathcal{Y}$ is said to be A.E. COMPUTABLE if it is partial computable with a measure-one domain.
- (3) (Hoyrup and Rojas [23]) A partial map $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is said to be an (A.E. COMPUTABLE) MORPHISM if it is a.e. computable and measure preserving, i.e. $\mu(T^{-1}(A)) = \nu(A)$ for all measurable $A \subseteq Y$.
- (4) (Hoyrup and Rojas [23]) A pair of partial maps $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ and $S: (\mathcal{Y}, \nu) \rightarrow (\mathcal{X}, \mu)$ are said to be an (A.E. COMPUTABLE) ISOMORPHISM if both maps are (a.e. computable) morphisms such that $(S \circ T)(x) = x$ for μ -a.e. $x \in \mathcal{X}$ and $(T \circ S)(y) = y$ for ν -a.e. $y \in \mathcal{Y}$. We also say $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is an isomorphism if such an S exists.

Note that this definition of isomorphism differs slightly from that of Hoyrup and Rojas [23]. They require that the domain must also be dense.

The definition of partial-computable map above basically says that the domain of T is determined by its algorithm and not some artificial restriction on the domain. This next proposition shows that this is equivalent to saying that the domain is Π_2^0 .

Proposition 7.2. *A partial map $T: \mathcal{X} \rightarrow \mathcal{Y}$ is partial computable if and only if the domain of T is a Π_2^0 set and T is computable on its domain.*

Proof. The proof of the first direction is straightforward. (For example, given $F: \mathbb{N}^\omega \rightarrow \mathbb{N}^\omega$, then $\text{dom}(F)$ is Π_2^0 in \mathbb{N}^ω [39, Theorem 2.2.4]. Also, the set of \mathcal{X} -Cauchy-names is Π_1^0 and the set of pairs (a, b) such that $a \approx_{\mathcal{X}} b$ (i.e. a and b are Cauchy-names for the same point in \mathcal{X}) and $h(a) \not\approx_{\mathcal{Y}} h(b)$ is Δ_2^0 .)

For the other direction, let D be the Π_2^0 domain. Then $D = \bigcap_n U_n$ where (U_n) is a computable sequence of Σ_1^0 sets. Let $F: \mathbb{N}^\omega \rightarrow \mathbb{N}^\omega$ be the partial-computable map from Cauchy-names to Cauchy-names that computes T . Modify $F(a)$ to return an n th approximation only if a “looks like” a Cauchy-name for some $x \in U_n$. \square

This next corollary says a.e. computable maps are defined on Kurtz randoms. Further, Kurtz randomness can be characterized by a.e. computable maps, and a.e. computable maps are determined by their values on Kurtz randoms. (For a different characterization of Kurtz randomness using a.e. computable functions, see Hertling and Yongge [20].)

Corollary 7.3. *Let (\mathcal{X}, μ) be a computable measure space and \mathcal{Y} a computable Polish space. For $x \in \mathcal{X}$, x is Kurtz random if and only if it is in the domain of every a.e. computable map $T: (\mathcal{X}, \mu) \rightarrow \mathcal{Y}$. Further, two a.e. computable maps are a.e. equal if and only if they agree on Kurtz randoms.*

Proof. For the first part, if x is Kurtz random, it avoids all null Σ_2^0 sets, and by Proposition 7.2 is in the domain of every a.e. computable map. Conversely, x is not Kurtz random, it is in some null Σ_2^0 set A . But the partial map $T: (\mathcal{X}, \mu) \rightarrow \mathcal{Y}$ with domain $\mathcal{X} \setminus A$ and $T(x) = 1$ for $x \in \mathcal{X} \setminus A$ is a.e. computable by Proposition 7.2.

For the second part, let $T, S: (\mathcal{X}, \mu) \rightarrow \mathcal{Y}$ be a.e. computable maps that are a.e. equal. The set $\{x \in \mathcal{X} \mid T(x) \neq S(x)\}$ is a null Σ_2^0 set in \mathcal{X} . Conversely, if $T(x) = S(x)$ for all Kurtz randoms x , then $T = S$ a.e. \square

Remark 7.4 (Preimages of Σ_1^0 sets). The preimage of a Σ_1^0 set under an computable map is still Σ_1^0 . Unfortunately, the preimage of a Σ_1^0 set under an *partial computable map* is not always Σ_1^0 . However, it is equal to the intersection of a Σ_1^0 set and the domain of the map. (We leave the details to the reader.) As an abuse of notation, if $T: \mathcal{X} \rightarrow \mathcal{Y}$ is a partial-computable map and $V \subseteq \mathcal{Y}$ is Σ_1^0 , we will define $T^{-1}(V)$ to be a Σ_1^0 set $U \subseteq \mathcal{X}$ such that for all $x \in \mathcal{X}$, $x \in U \cap \text{dom}(T) \Leftrightarrow T(x) \in V$. (We leave it to the reader to verify that such a U can be computed uniformly from the codes for T and V .) Also, if $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is a morphism, it is easy to see that $\mu(U) = \nu(V)$. We can similarly define the preimage of a $\Pi_1^0, \Sigma_2^0, \Pi_2^0$ set to remain in the same point class. Last, if $B \subseteq \mathcal{Y}$ is a.e. decidable with a.e. decidable pair (V_0, V_1) , then define $T^{-1}(B)$ to be the a.e. decidable set A given by the pair $(T^{-1}(V_0), T^{-1}(V_1))$.

This next proposition shows that for many common notions of randomness are preserved by morphisms, and the set of randoms is preserved under isomorphisms.

Proposition 7.5. *If $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is a morphism and $x \in \mathcal{X}$ is Martin-Löf random, then $T(x)$ is Martin-Löf random. The same is true of Kurtz and Schnorr randomness. Hence, if T is an isomorphism, then x is Martin-Löf (respectively Kurtz, Schnorr) random if and only if $T(x)$ is.*

Proof. Assume $T(x)$ is not Martin-Löf random in (\mathcal{Y}, ν) . Then there is a Martin-Löf test (U_n) in (\mathcal{Y}, ν) which covers $T(x)$. Let $V_n = T^{-1}(U_n)$ for each n . By Remark 7.4 (V_n) is a Martin-Löf test in (\mathcal{X}, μ) which covers x . Hence x is not Martin-Löf random in (\mathcal{X}, μ) .

Kurtz and Schnorr randomness follow similarly, namely the inverse image of a test is still a test. \square

(Bienvenu and Porter have pointed out to me the following partial converse to Proposition 7.5, which was first proved by Shen—see [7]. If $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is a morphism and y is Martin-Löf random for (\mathcal{Y}, ν) , then there is some x that is Martin-Löf random for (\mathcal{X}, μ) such that $T(x) = y$.)

In Corollary 9.7, we will see that computable randomness is not preserved by morphisms. However, just looking at the previous proof gives a clue. There is another criterion to the tests for computable randomness besides complexity and measure, namely the cell decompositions of the space. The “inverse image” of cell decomposition may not be a cell decomposition.

However, if T is an isomorphism the situation is much better. Indeed, these next three propositions show a correspondence between isomorphisms and cell decompositions. We say two cell decompositions \mathcal{A} and \mathcal{B} of a computable probability space (\mathcal{X}, μ) are almost-everywhere equal if $[\sigma]_{\mathcal{A}} = [\sigma]_{\mathcal{B}}$ a.e. for all $\sigma \in 2^{<\omega}$. Recall, two isomorphisms are almost-everywhere equal if they are pointwise a.e. equal.

Proposition 7.6 (Isomorphisms to cell decompositions). *If $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ is an isomorphism and \mathcal{B} is a cell decomposition of (\mathcal{Y}, ν) , then there is an a.e. unique cell decomposition \mathcal{A} (which we will notate as $T^{-1}(\mathcal{B})$) such that $\text{name}_{\mathcal{A}}(x) = \text{name}_{\mathcal{B}}(T(x))$ for μ -a.e. x . This cell decomposition \mathcal{A} is given by $[\sigma]_{\mathcal{A}} = T^{-1}([\sigma]_{\mathcal{B}})$. In particular, every isomorphism $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$ induces a cell decomposition \mathcal{A} such that $\text{name}_{\mathcal{A}}(x) = T(x)$ for μ -a.e. x .*

Proof. We will show $[\sigma]_{\mathcal{A}} = T^{-1}([\sigma]_{\mathcal{B}})$ defines a cell decomposition \mathcal{A} . By Remark 7.4, $T^{-1}([\sigma]_{\mathcal{B}})$ is Σ_1^0 uniformly from σ . Clearly, $\mu([\varepsilon]_{\mathcal{A}}) = 1$, $[\sigma 0]_{\mathcal{A}} \cap [\sigma 1]_{\mathcal{A}} = \emptyset$, and $[\sigma 0]_{\mathcal{A}} \cup [\sigma 1]_{\mathcal{A}} = [\sigma]_{\mathcal{A}}$ μ -a.e. Finally, take a Σ_1^0 set $U \subseteq \mathcal{X}$. By Proposition 4.8, it is enough to show there is some c.e. set $\{\sigma_i\}$ such that $U = \bigcup_i [\sigma_i]_{\mathcal{A}}$ μ -a.e. Let S be the inverse isomorphism to T . Then define $V = S^{-1}(U)$. By Remark 7.4, V is Σ_1^0 in \mathcal{Y} and $T^{-1}(V) = U$ μ -a.e. By Proposition 4.7 there is some c.e. set $\{\sigma_i\}$ such that $V = \bigcup_i [\sigma_i]_{\mathcal{B}}$ ν -a.e. and therefore $U = T^{-1}(V) = \bigcup_i T^{-1}([\sigma_i]_{\mathcal{B}}) = \bigcup_i [\sigma_i]_{\mathcal{A}}$ μ -a.e. Therefore, $[\sigma]_{\mathcal{A}} = T^{-1}([\sigma]_{\mathcal{B}})$ defines a cell decomposition \mathcal{A} .

For μ -a.e. x , $x \in \text{dom}(T) \cap \text{dom}(\text{name}_{\mathcal{A}})$. Then for all n , $x \in [x \upharpoonright_{\mathcal{A}} n]_{\mathcal{A}} = T^{-1}([x \upharpoonright_{\mathcal{A}} n]_{\mathcal{B}})$. By Remark 7.4, $T(x) \in [x \upharpoonright_{\mathcal{A}} n]_{\mathcal{B}}$. Therefore $\text{name}_{\mathcal{B}}(T(x)) = \text{name}_{\mathcal{A}}(x)$.

For $\mathcal{Y} = 2^\omega$, let \mathcal{B} be the natural cell decomposition of $(2^\omega, \nu)$, then $[\sigma]_{\mathcal{B}} = [\sigma]^\prec$ for all $\sigma \in 2^{<\omega}$. Therefore for μ -a.e. x , $\text{name}_{\mathcal{A}}(x) = \text{name}_{\mathcal{B}}(T(x)) = T(x)$.

To show the cell decomposition \mathcal{A} is unique, assume \mathcal{A}' is another cell decomposition such that for μ -a.e. x , the \mathcal{A} -name and \mathcal{A}' -name of x are both the \mathcal{B} name of $T(x)$. Then $[\sigma]_{\mathcal{A}} = [\sigma]_{\mathcal{A}'}$ μ -a.e. for all $\sigma \in 2^{<\omega}$. \square

Proposition 7.7 (Cell decompositions to isomorphisms). *Let (\mathcal{X}, μ) be a computable probability space with cell decomposition \mathcal{A} . There is a unique computable probability space $(2^\omega, \mu_{\mathcal{A}})$ such that $\text{name}_{\mathcal{A}}: (\mathcal{X}, \mu) \rightarrow (2^\omega, \mu_{\mathcal{A}})$ is an isomorphism. Namely, $\mu_{\mathcal{A}}(\sigma) = \mu([\sigma]_{\mathcal{A}})$.*

Proof. If such a measure $\mu_{\mathcal{A}}$ exists, it must be unique. Indeed, since $\text{name}_{\mathcal{A}}$ is then measure-preserving, $\mu_{\mathcal{A}}$ must satisfy $\mu_{\mathcal{A}}(\sigma) = \mu(\text{name}_{\mathcal{A}}^{-1}([\sigma]_{\mathcal{A}})) = \mu([\sigma]_{\mathcal{A}})$, which uniquely defines $\mu_{\mathcal{A}}$.

It remains to show the map $\text{name}_{\mathcal{A}}: (\mathcal{X}, \mu) \rightarrow (2^\omega, \mu_{\mathcal{A}})$ which maps x to its \mathcal{A} -name is an isomorphism. Clearly, $\mu_{\mathcal{A}}$ is a computable measure since $\mu([\sigma]_{\mathcal{A}})$ is computable. The map $\text{name}_{\mathcal{A}}$ which takes x to its \mathcal{A} -name is measure preserving for cylinder sets and therefore for all sets by approximation. The map from x to $x \upharpoonright_{\mathcal{A}} n$ is a.e. computable. Indeed, wait for x to show up in one of the sets $[\sigma]_{\mathcal{A}}$ where $|\sigma| = n$. Hence the map from x to its \mathcal{A} -name is also a.e. computable. So $\text{name}_{\mathcal{A}}$ is a morphism. (As an extra verification, clearly $\text{dom}(\text{name}_{\mathcal{A}})$ is a Π_2^0 measure-one set.)

The inverse of $\text{name}_{\mathcal{A}}$ will be the map S from (a measure-one set of) \mathcal{A} -names $y \in 2^\omega$ to points $x \in \mathcal{X}$ such that $\text{name}_{\mathcal{A}}(x) = y$. The algorithm for S will be similar to the algorithm given by the proof of the Baire category theorem (see [10]). Pick $y \in 2^\omega$. We compute $S(y)$ by a back-and-forth argument. Assume $\tau \prec y$. Recall, $[\tau]_{\mathcal{A}}$ is Σ_1^0 . We can enumerate a sequence of pairs (a_i, k_i) where each a_i is a simple point of \mathcal{X} and each $k_i > |\tau|$ such that $[\tau]_{\mathcal{A}} = \bigcup_i B(a_i, 2^{-k_i})$. Further, by Proposition 4.7, we have that for each i , there is a c.e. set $\{\sigma_j^i\}$ such that $B(a_i, 2^{-k_i}) = \bigcup_j [\sigma_j^i]_{\mathcal{A}}$ μ -a.e. (We may assume $|\sigma_j^i| > |\tau|$ for all i, j .) Given y , compute the Cauchy-name of $S(y)$ as follows. Start with $\tau_1 = y \upharpoonright 1$. Then search for $\sigma_j^i \prec y$. If we find one, let $b_1 = a_i$ be the first approximation. Now continue with $\tau_2 = \sigma_j^i$, and so on. This gives a Cauchy-name (b_n) . The algorithm will fail if at some stage it cannot find any $\sigma_j^i \prec y$. But then $y \in [\tau]_{\mathcal{A}} \setminus \bigcup_i \bigcup_j [\sigma_j^i]_{\mathcal{A}}$, which by the definition of $\mu_{\mathcal{A}}$, is a $\mu_{\mathcal{A}}$ -measure-zero set since $[\tau]_{\mathcal{A}} = \bigcup_i \bigcup_j [\sigma_j^i]_{\mathcal{A}}$ μ -a.e. Hence S is a.e. computable.

By the back-and-forth algorithm, $\text{name}_{\mathcal{A}}(S(y)) = y$ for all $y \in \text{dom}(S)$. To show $S(\text{name}_{\mathcal{A}}(x)) = x$ a.e., assume $x \in \text{dom}(\text{name}_{\mathcal{A}})$. Consider the back-and-forth sequence created by the algorithm: $[\tau_n]_{\mathcal{A}} \supseteq B(b_n, 2^{-k_n}) \supseteq [\tau_{n+1}]_{\mathcal{A}} \supseteq \dots$. For all n , we have $\tau_n \prec \text{name}_{\mathcal{A}}(x)$, then $x \in [\tau_n]_{\mathcal{A}}$ for all n . So $x = \lim_{n \rightarrow \infty} b_n = S(\text{name}_{\mathcal{A}}(x))$. Since $S^{-1}([\sigma]_{\mathcal{A}}) = S^{-1}(\text{name}_{\mathcal{A}}^{-1}([\sigma]_{\mathcal{A}})) = [\sigma]_{\mathcal{A}}$ $\mu_{\mathcal{A}}$ -a.e., S is a measure-preserving map, and hence a morphism. Therefore, $\text{name}_{\mathcal{A}}$ is an isomorphism. \square

These last two propositions show that there is a one-to-one correspondence between cell decompositions \mathcal{A} of a space (\mathcal{X}, μ) and isomorphisms of the form $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$. This next proposition shows a further one-to-one correspondence between isomorphisms $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ and $S: (2^\omega, \mu_{\mathcal{A}}) \rightarrow (2^\omega, \nu_{\mathcal{B}})$.

Proposition 7.8 (Pairs of cell decompositions to isomorphisms). *Let (\mathcal{X}, μ) and (\mathcal{Y}, ν) be computable probability spaces with cell decompositions \mathcal{A} and \mathcal{B} . Let $\mu_{\mathcal{A}}$ be as in Proposition 7.7, and similarly for $\nu_{\mathcal{B}}$. Then for every isomorphism $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ there is an a.e. unique isomorphism $S: (2^\omega, \mu_{\mathcal{A}}) \rightarrow (2^\omega, \nu_{\mathcal{B}})$ and vice versa, such that S maps $\text{name}_{\mathcal{A}}(x)$ to $\text{name}_{\mathcal{B}}(T(x))$ for μ -a.e. $x \in \mathcal{X}$. In*

other words the following diagram commutes for μ -a.e. $x \in \mathcal{X}$.

$$\begin{array}{ccc} (\mathcal{X}, \mu) & \xrightarrow{\text{name}_{\mathcal{A}}} & (2^\omega, \mu_{\mathcal{A}}) \\ \downarrow T & & \downarrow S \\ (\mathcal{Y}, \nu) & \xrightarrow{\text{name}_{\mathcal{B}}} & (2^\omega, \nu_{\mathcal{B}}) \end{array}$$

Further we have the following.

- (1) If (\mathcal{X}, μ) equals (\mathcal{Y}, ν) , then T is the identity isomorphism precisely when S is the isomorphism which maps $\text{name}_{\mathcal{A}}(x)$ to $\text{name}_{\mathcal{B}}(x)$.
- (2) Conversely, S is the identity isomorphism (and hence $\mu_{\mathcal{A}}$ equals $\nu_{\mathcal{B}}$) precisely when $\mathcal{A} = T^{-1}(\mathcal{B})$ (as in Proposition 7.6).

Proof. Given T , let $S = \text{name}_{\mathcal{A}}^{-1} \circ T \circ \text{name}_{\mathcal{B}}$, and similarly to get T from S . Then the diagram clearly commutes. A.e. uniqueness follows since the maps are isomorphisms.

If $\mathcal{A} = T^{-1}(\mathcal{B})$ is induced by T , then by Proposition 7.6, $\text{name}_{\mathcal{A}}(x) = \text{name}_{\mathcal{B}}(T(x))$ which makes S the identity map. But since S is an isomorphism, $\mu_{\mathcal{A}}$ and $\nu_{\mathcal{B}}$ must be the same measure.

The rest follows from “diagram chasing”. \square

Now we can show computable randomness is preserved by isomorphisms.

Theorem 7.9. *Isomorphisms preserve computable randomness. Namely, given an isomorphism $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$, then $x \in \mathcal{X}$ is computably random if and only if $T(x)$ is.*

Proof. Assume $T(x)$ is not computably random. Fix an isomorphism $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$. Let \mathcal{B} be a cell decomposition of (\mathcal{Y}, ν) . Take a bounded Martin-Löf test (U_n) on (\mathcal{Y}, ν) with bounding measure κ with respect to \mathcal{B} which covers $T(x)$. By Proposition 7.6 there is a cell decomposition $\mathcal{A} = T^{-1}(\mathcal{B})$ on (\mathcal{X}, μ) such that $[\sigma]_{\mathcal{A}} = T^{-1}([\sigma]_{\mathcal{B}})$ for all $\sigma \in 2^{<\omega}$. Define $V_n = T^{-1}(U_n)$. Then (V_n) is a bounded Martin-Löf test on (\mathcal{X}, μ) bounded by the same measure κ with respect to \mathcal{A} . Indeed,

$$\mu(V_n \cap [\sigma]_{\mathcal{A}}) = \nu(U_n \cap [\sigma]_{\mathcal{B}}) \leq 2^{-n} \kappa(\sigma).$$

Also, (V_n) covers x , hence x is not computably random. \square

Using Theorem 7.9, we can explore computable randomness on various spaces.

Example 7.10 (Computably random vectors). Let $([0, 1]^d, \lambda)$ be the cube $[0, 1]^d$ with the Lebesgue measure. The following is a natural isomorphism from $([0, 1]^d, \lambda)$ to $(2^\omega, \lambda)$. First, represent $(x_1, \dots, x_d) \in [0, 1]^d$ by the binary sequence of each component; then interleave the binary sequences. By Theorem 7.9, (x_1, \dots, x_d) is computably random if and only if the sequence of interleaved binary sequences is computably random. (This definition of computable randomness on $[0, 1]^d$ was proposed by Brattka, Miller and Nies [12].)

Example 7.11 (Base invariance). Let λ_3 be the uniform measure on 3^ω . Consider the natural isomorphism $T_{2,3}: (2^\omega, \lambda) \rightarrow (3^\omega, \lambda_3)$ which identifies the binary and ternary expansions of a real. This is an a.e. computable isomorphism, so $x \in [0, 1]$ is computably random if and only if $T_{2,3}(x)$ is computably random. We say a randomness notion (defined on (b^ω, λ_b) for all $b \geq 2$, see Example 5.8) is **BASE INVARIANT** if this property holds for all base pairs b_1, b_2 .

Example 7.12 (Computably random Brownian motion). Consider the space $C([0, 1])$ of continuous functions from $[0, 1]$ to \mathbb{R} endowed with the Wiener probability measure W (i.e. the measure of Brownian motion). The space $C([0, 1])$ is a computable Polish space (where the simple points are the rational piecewise linear functions). Fouché [15] gave an computable measure preserving map Φ which takes as an input a sequence of fair-coin flips and returns (the code for) a Brownian motion in $C([0, 1])$. He also showed the corresponding inverse map is a.e. computable. Therefore Φ is an a.e. computable bijection between a measure-one set of $(2^\omega, \lambda)$ and of $(C([0, 1]), W)$. This implies that $C([0, 1], W)$ is a computable probability space and $\Phi: (2^\omega, \lambda) \rightarrow (C([0, 1]), W)$ is an a.e. computable isomorphism. Hence the computably random Brownian motions (i.e. the computably random points of $(C([0, 1]), W)$) are exactly the functions which arise from Fouché's construction using a computably random sequence from $(2^\omega, \lambda)$.

Example 7.13 (Computably random closed set). Consider the space $\mathcal{F}(2^\omega)$ of closed sets of 2^ω . This space has a topology called the Fell topology. The subspace $\mathcal{F}(2^\omega) \setminus \{\emptyset\}$ can naturally be identified with trees on $\{0, 1\}$ with no dead branches. Barmpalias et al. [3] gave a natural construction of these trees from ternary strings in 3^ω . Axon [2] showed the corresponding map $T: 3^\omega \rightarrow \mathcal{F}(2^\omega) \setminus \{\emptyset\}$ is a homeomorphism between 3^ω and the Fell topology restricted to $\mathcal{F}(2^\omega) \setminus \{\emptyset\}$. Hence $\mathcal{F}(2^\omega) \setminus \{\emptyset\}$ can be represented as a computable Polish space, and the probability measure on $\mathcal{F}(2^\omega) \setminus \{\emptyset\}$ induced by T can be represented as a computable probability measure. Since T is an a.e. computable isomorphism, the computably random closed sets of this space are then the ones whose corresponding trees are constructed from computably random ternary strings in 3^ω .

Example 7.14 (Computably random structures). The last two examples can be extended to a number of random structures—infinite random graphs, Markov processes, random walks, random matrices, Galton-Watson processes, etc. The main idea is as follows. Assume (Ω, P) is a computable probability space (the sample space), \mathcal{X} is the space of structures, and $T: (\Omega, P) \rightarrow \mathcal{X}$ is an a.e. computable map (a random structure). This induces a measure μ on \mathcal{X} (the distribution of T). If, moreover, T is an a.e. computable isomorphism between (Ω, P) and (\mathcal{X}, μ) , then the computably random structures of (\mathcal{X}, μ) are exactly the objects constructed from computably random points in (Ω, P) .

In this next theorem, an ATOM (or POINT-MASS) is a point with positive measure. An ATOMLESS probability space is one without atoms.

Theorem 7.15 (Hoyrup and Rojas [23]). *If (\mathcal{X}, μ) is an atomless computable probability space, then there is a isomorphism $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$. Further, T is computable from (\mathcal{X}, μ) .*

Corollary 7.16. *If (\mathcal{X}, μ) is an atomless computable probability space, then $x \in \mathcal{X}$ is computably random if and only if $T(x)$ is computably random for any (and all) isomorphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$.*

Proof. Follows from Theorems 7.9 and 7.15. □

Corollary 7.17. *Given a measure (\mathcal{X}, μ) with cell decomposition \mathcal{A} , $x \in X$ is computably random if and only if $\text{name}_{\mathcal{A}}(x)$ is computably random in $(2^\omega, \mu_{\mathcal{A}})$ where $\mu_{\mathcal{A}}(\sigma) = \mu([\sigma]_{\mathcal{A}})$.*

Proof. Use Proposition 7.7 and Theorem 7.9. □

8. GENERALIZING RANDOMNESS TO COMPUTABLE PROBABILITY SPACES

In this section, I explain the general method of this paper which generalizes a randomness notion from $(2^\omega, \lambda)$ to an arbitrary computable measure space.

Imagine we have an arbitrary randomness notion called X -RANDOMNESS defined on $(2^\omega, \lambda)$. (Here X is a place-holder for a name like “Schnorr” or “computable”; it has no relation to being random relative to an oracle.) The definition of X -random should either explicitly or implicitly assume we are working in the fair-coin measure. The method can be reduced to three steps.

Step 1: Generalize X -randomness to computable probability measures on 2^ω . This is self-explanatory, although not always trivial.

Step 2: Generalize X -randomness to computable probability spaces. There are three equivalent ways to do this for a computable probability space (\mathcal{X}, μ) .

- (1) Replace all instances of $[\sigma]^\prec$ with $[\sigma]_{\mathcal{A}}$, $x \upharpoonright n$ with $x \upharpoonright_{\mathcal{A}} n$, etc. in the definition of X -random from Step 1. Call this X^* -random with respect to \mathcal{A} . Then define $x \in \mathcal{X}$ to be X^* -random on (\mathcal{X}, μ) if it is X^* -random with respect to all cell decompositions \mathcal{A} (ignoring unrepresented points of \mathcal{A} and points in null cells—which are not even Kurtz random). (Compare with Definition 5.4.)
- (2) Define $x \in \mathcal{X}$ to be X^* -random on (\mathcal{X}, μ) if for each cell decomposition \mathcal{A} , $\text{name}_{\mathcal{A}}(x)$ is X -random on $(2^\omega, \mu_{\mathcal{A}})$, where $\mu_{\mathcal{A}}$ is given by $\mu_{\mathcal{A}}(\sigma) = \mu([\sigma]_{\mathcal{A}})$. (Compare with Corollary 7.17.)
- (3) Define $x \in \mathcal{X}$ to be X^* -random on (\mathcal{X}, μ) if for all isomorphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$ we have that $T(x)$ is X -random on $(2^\omega, \nu)$. (Compare with Theorem 7.9.)

The description of (1) is a bit vague, but when done correctly it is the most useful definition. The definition given by (1) should be equivalent to that given by (2) because (1) is essentially about \mathcal{A} -names. To see that (2) and (3) give the same definition, use Propositions 7.6 and 7.7, which show that isomorphisms to 2^ω are maps to \mathcal{A} -names and vice versa.

Step 3: Verify that the new definition is consistent with the original. It may be that on $(2^\omega, \lambda)$ the class of X^* -random points is strictly smaller than the class of the original X -random points. There are three equivalent techniques to show that X^* -randomness on 2^ω is equivalent to X -randomness. The three techniques are related to the three definitions from Step 2.

- (1) Show the definition of X^* -randomness is invariant under the choice of cell decomposition. (Compare with Theorem 5.6.)
- (2) Show that for every two cell decompositions \mathcal{A} and \mathcal{B} , the \mathcal{A} -name of x is X -random on $(2^\omega, \mu_{\mathcal{A}})$ if and only if the \mathcal{B} -name is X -random on $(2^\omega, \mu_{\mathcal{B}})$. (Compare with Corollary 7.17.)
- (3) Show that X -randomness is invariant under all isomorphisms from $(2^\omega, \mu)$ to $(2^\omega, \nu)$. (Compare with Theorem 7.9.)

Again, these three approaches are equivalent. Assuming the definition is stated correctly, (1) and (2) say the same thing.

To see that (3) implies (2), assume X -randomness is invariant under isomorphisms on 2^ω . Consider two cell decompositions \mathcal{A} and \mathcal{B} of the same space (\mathcal{X}, μ) .

By Proposition 7.8 (1), there is an isomorphism $S: (2^\omega, \mu_{\mathcal{A}}) \rightarrow (2^\omega, \mu_{\mathcal{B}})$ which maps \mathcal{A} -names to \mathcal{B} -names, i.e. this diagram commutes.

$$\begin{array}{ccc} (\mathcal{X}, \mu) & \xrightarrow{\text{name}_{\mathcal{A}}} & (2^\omega, \mu_{\mathcal{A}}) \\ & \searrow \text{name}_{\mathcal{B}} & \downarrow S \\ & & (2^\omega, \mu_{\mathcal{B}}) \end{array}$$

Since S preserves \mathbf{X} -randomness, $\text{name}_{\mathcal{A}}(x)$ is \mathbf{X} -random on $(2^\omega, \mu_{\mathcal{A}})$ if and only if and only if $\text{name}_{\mathcal{B}}(x)$ is \mathbf{X} -random on $(2^\omega, \mu_{\mathcal{B}})$.

To see that (2) implies (3), assume that (2) holds. Consider an isomorphism $S: (2^\omega, \mu) \rightarrow (2^\omega, \nu)$. Let (\mathcal{X}, κ) be any space isomorphic to $(2^\omega, \mu)$. Then (\mathcal{X}, κ) is also isomorphic to $(2^\omega, \nu)$. So there are isomorphisms T_1 and T_2 such that this diagram commutes.

$$\begin{array}{ccc} (\mathcal{X}, \kappa) & \xrightarrow{T_1} & (2^\omega, \mu) \\ & \searrow T_2 & \downarrow S \\ & & (2^\omega, \nu) \end{array}$$

By Proposition 7.6 there are two cell decompositions \mathcal{A} and \mathcal{B} on (\mathcal{X}, κ) such that $T_1 = \text{name}_{\mathcal{A}}$ and $(2^\omega, \mu) = (2^\omega, \kappa_{\mathcal{A}})$. The same holds for \mathcal{B} and ν . Then we have this commutative diagram.

$$\begin{array}{ccc} (\mathcal{X}, \kappa) & \xrightarrow{\text{name}_{\mathcal{A}}} & (2^\omega, \kappa_{\mathcal{A}}) \\ & \searrow \text{name}_{\mathcal{B}} & \downarrow S \\ & & (2^\omega, \kappa_{\mathcal{B}}) \end{array}$$

Consider any \mathbf{X} -random $y \in (2^\omega, \kappa_{\mathcal{A}})$. It is the \mathcal{A} -name of some $x \in (\mathcal{X}, \kappa)$, in other words $y = \text{name}_{\mathcal{A}}(x)$. By (2), we also have that $\text{name}_{\mathcal{B}}(x)$ is \mathbf{X} -random. So S preserves \mathbf{X} -randomness.

Notice that Step 3 implies that some randomness notions cannot be generalized without making the set of randoms smaller. This is because they are not invariant under isomorphisms between computable probability measures on 2^ω . Yet, even when the \mathbf{X}^* -randoms are a proper subclass of the \mathbf{X} -randoms, the \mathbf{X}^* randoms are an interesting class of randomness. In particular we have the following.

Proposition 8.1. *\mathbf{X}^* -randomness is invariant under isomorphisms.*

In some sense the \mathbf{X}^* -randoms are the largest such subclass of the \mathbf{X} -randoms. (One must be careful how to say this, since \mathbf{X} -randomness is only defined on measures $(2^\omega, \mu)$.)

Proof. Let $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{Y}, \nu)$ be an isomorphism and let $x \in (\mathcal{X}, \mu)$ be \mathbf{X}^* -random. Let \mathcal{B} be a arbitrary cell decomposition of (\mathcal{Y}, ν) . Since \mathcal{B} is arbitrary, it is enough

to show that $\text{name}_B(T(x))$ is \mathbf{X} -random in $(2^\omega, \nu_B)$. By Proposition 7.6 and Proposition 7.8 (2) we have a cell decomposition \mathcal{A} on (\mathcal{X}, μ) such that $(2^\omega, \mu_A) = (2^\omega, \nu_B)$ and the following diagram commutes.

$$\begin{array}{ccc} (\mathcal{X}, \mu) & \xrightarrow{\text{name}_A} & (2^\omega, \mu_A) \\ \downarrow T & & \parallel \\ (\mathcal{Y}, \nu) & \xrightarrow{\text{name}_B} & (2^\omega, \nu_B) \end{array}$$

Since $x \in (\mathcal{X}, \mu)$ is \mathbf{X}^* -random, $\text{name}_A(x)$ is \mathbf{X} -random in $(2^\omega, \mu_A) = (2^\omega, \nu_B)$. Since the diagram commutes, $\text{name}_B(T(x))$ is also \mathbf{X} -random in $(2^\omega, \nu_B)$. Since \mathcal{B} is arbitrary, x is \mathbf{X} -random. \square

In the case that (\mathcal{X}, μ) is an atomless computable probability measure, we could instead define $x \in \mathcal{X}$ to be \mathbf{X}^\star -random if $T(x)$ is random for all isomorphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$. We can then skip Step 1, and in Step 3 it is enough to check that \mathbf{X} -randomness is invariant under automorphisms of $(2^\omega, \lambda)$. Similarly, \mathbf{X}^\star -randomness would be invariant under isomorphisms.

9. BETTING STRATEGIES AND KOLMOGOROV-LOVELAND RANDOMNESS

In the next two sections I consider how the method of Section 8 can be applied to Kolmogorov-Loveland randomness, which is also defined through a betting strategy on the bits of the string.

Call a betting strategy on bits **NONMONOTONIC** if the gambler can decide at each stage which coin flip to bet on. For example, maybe the gambler first bets on the 5th bit. If it is 0, then he bets on the 3rd bit; if it is 1, he bets on the 8th bit. (Here, and throughout this paper we still assume the gambler cannot bet more than what is in his capital, i.e. he cannot take on debt.) A string $x \in 2^\omega$ is **KOLMOGOROV-LOVELAND RANDOM** or **NONMONOTONICALLY RANDOM** (in $(2^\omega, \lambda)$) if there is no computable nonmonotonic betting strategy on the bits of the string which succeeds on x .

Indeed, this gives a lot more freedom to the gambler and leads to a strictly stronger notion than computable randomness. While it is easy to show that every Martin-Löf random is Kolmogorov-Loveland random, the converse is a difficult open question.

Question 9.1. *Is Kolmogorov-Loveland randomness the same as Martin-Löf randomness?*

On one hand, there are a number of results that show Kolmogorov-Loveland randomness is very similar to Martin-Löf randomness. On the other hand, it is not even known if Kolmogorov-Loveland randomness is base invariant, and it is commonly thought that Kolmogorov-Loveland randomness is strictly weaker than Martin-Löf randomness. For the most recent results on Kolmogorov-Loveland randomness see [14, Section 7.5], [32, Section 7.6], and [5, 24, 29].

In this section I will ask what type of randomness one gets by applying the method of Section 8 to Kolmogorov-Loveland randomness. The result is Martin-Löf randomness. However, this does not prove that Kolmogorov-Loveland randomness is the same as Martin-Löf randomness, since I leave as an open question whether

Kolmogorov-Loveland randomness (naturally extended to all computable probability measures on 2^ω) is invariant under isomorphisms. The presentation of this section follows the three-step method of Section 8.

9.1. Step 1: Generalize to other computable probability measures μ on 2^ω . Kolmogorov-Loveland randomness can be naturally extended to computable probability measures on 2^ω . Namely, bet as usual, but adjust the payoffs to be fair. For example, if the gambler wagers 1 unit of money to bet that $x(4) = 1$ (i.e. the 4th bit is 1) after seeing that $x(2) = 1$ and $x(6) = 0$, then if he wins, the fair payoff is

$$\frac{\mu(x(4) = 0 \mid x(2) = 1, x(6) = 0)}{\mu(x(4) = 1 \mid x(2) = 1, x(6) = 0)}.$$

where $\mu(A \mid B) = \mu(A \cap B) / \mu(B)$ represents the conditional probability of A given B . If the gambler loses, he loses his unit of money.

(Note, we could also allow the gambler to bet on a bit he has already seen. Indeed, he will not win any money. This would, however, introduce “partial randomness” since the gambler could delay betting on a new bit. Nonetheless, Merkle [27] showed that partial Kolmogorov-Loveland randomness is the same as Kolmogorov-Loveland randomness.)

As with computable randomness, we must address division by zero. The gambler is not allowed to bet on a bit if it has probability zero of occurring (conditioned on the information already known). Instead we just declare the elements of such null cylinder sets to be not random.

9.2. Step 2: Generalize Kolmogorov-Loveland randomness to computable probability measures. Pick a computable probability measure (\mathcal{X}, μ) with generator $\mathcal{A} = (A_n)$. Following the second step of the method in Section 8, the gambler bets on the bits of the \mathcal{A} -name of x . A little thought reveals that what the gambler is doing when she bets that the n th bit of the \mathcal{A} -name is 1 is betting that $x \in A_n$. For any generator \mathcal{A} , if we add more a.e. decidable sets to \mathcal{A} , it is still a generator. Further, since we are not necessarily betting on all the sets in \mathcal{A} , we do not even need to assume \mathcal{A} is anything more than a collection of a.e. decidable sets. (This is the key difference between computable randomness.)

Hence, we may think of the betting strategy as follows. The gambler chooses some a.e. decidable set A and bets that $x \in A$ (or x has property A). (Again, the gambler must know that $\mu(A) > 0$ before betting on it.) Then if she wins, she gets a fair payout, and if she loses, she loses her bet. Call such a strategy a **COMPUTABLE BETTING STRATEGY**. Call the resulting randomness **BETTING RANDOMNESS**. (A more formal definition is given in Remark 9.4.)

I argue that betting randomness is the most general randomness notion that can be described by a finitary fair-game betting scenario with a “computable betting strategy.” Indeed, consider these three basic properties of such a game:

- (1) The gambler must be able determine (almost surely) some property of x that she is betting on, and this determination must be made with only the information about x that she has gained during the game.
- (2) A bookmaker must be able to determine (almost surely) if this property holds of x or not.
- (3) If the gambler wins, the bookmaker must be able to determine (almost surely) the fair payoff amount.

The only way to satisfy (2) is if the property is a.e. decidable. Then (3) follows since a.e. decidable sets have finite descriptions and their measures are computable. To satisfy (1), the gambler must be able to compute the a.e. decidable set only knowing the results of her previous bets. This is exactly the computable betting strategy defined above.¹

Now recall Schnorr’s Critique that Martin-Löf randomness does not have a “computable-enough” definition. The definition Schnorr had in mind was a betting scenario. In particular, Schnorr gave a martingale characterization of Martin-Löf randomness that is the same as that of computable randomness, except the martingales are only lower semicomputable [35] (see also [14, 32]). If Martin-Löf randomness equals Kolmogorov-Loveland randomness, then some believe that this will give a negative answer to Schnorr’s Critique; namely, we will have found a computable betting strategy that describes Martin-Löf randomness. While, there is some debate as to what Schnorr meant by his critique (and whether he still agrees with it), we think the following is a worthwhile question.

Can Martin-Löf randomness be characterized using a finitary fair-game betting scenario with a “computable betting strategy”?

The answer turns out to be yes. As this next theorem shows, betting randomness is equivalent to Martin-Löf randomness. Hitchcock and Lutz [22] defined a generalization of martingales (as in the type used to define computable randomness on 2^ω) called martingale processes. In the terminology of this paper, a MARTINGALE PROCESS is basically a computable betting strategy on 2^ω with the fair-coin measure which bets on decidable sets (i.e. finite unions of basic open sets). Merkle, Mihailović and Slaman [28] showed that Martin-Löf randomness is equivalent to the randomness characterized by martingale processes. The proof of this next theorem is basically the Merkle et al. proof.²

Theorem 9.2. *Betting randomness and Martin-Löf randomness are the same.*

Proof. Fix a computable probability space (\mathcal{X}, μ) . To show Martin-Löf randomness implies betting randomness, we use a standard argument which was employed by Hitchcock and Lutz [22] for martingale processes. Assume $x \in \mathcal{X}$ is not betting random. Namely, there is some computable betting strategy \mathcal{B} which succeeds on x . Without loss of generality, the starting capital of \mathcal{B} may be assumed to be 1. Let $U_n = \{x \in \mathcal{X} \mid \mathcal{B} \text{ wins at least } 2^n \text{ on } x\}$. Each U_n is uniformly Σ_1^0 in n , and by a standard result in martingale theory $\mu(U_n) \leq C2^{-n}$ where $C = 1$ is the starting

¹In the three properties we did not consider the possibility of betting on a collection of three or more pair-wise disjoint events simultaneously. This is not an issue since one may break up the betting and bet on each event individually (see Example 5.8). There is also a more general possibility of having a computable or a.e. computable wager function over the space \mathcal{X} . This can be made formal using the martingales in probability theory, but it turns out that it does not change the randomness characterized by such a strategy. By an unpublished result of Ed Dean [personal communication], any L^1 -bounded layerwise-computable martingale converges on Martin-Löf randomness (which, as we will see, is equivalent to betting randomness).

²Downey and Hirschfeld [14, footnote on p. 269] also remark that the Merkle et al. result gives a possible answer to Schnorr’s critique.

capital.³ Hence (U_n) is a Martin-Löf test which covers x , and x is not Martin-Löf random.

For the converse, the argument is basically the Merkle, Mihailović and Slaman [28] proof for martingale processes.

First, let us prove a fact. Assume a gambler starts with a capital of 1 and $U \subset \mathcal{X}$ is some Σ_1^0 set such that $\mu(U) \leq 1/2$. Then there is a computable way that the gambler can bet on an unknown $x \in \mathcal{X}$ such that he doubles his capital (to 2) if $x \in U$ (actually, some Σ_1^0 set a.e. equal to U). The strategy is as follows. Choose a cell decomposition \mathcal{A} of (\mathcal{X}, μ) . Since U is Σ_1^0 , by Proposition 4.7 there is a c.e., prefix-free set $\{\sigma_i\}$ of finite strings such that $U = \bigcup_i [\sigma_i]_{\mathcal{A}}$ a.e. We may assume $\mu([\sigma_i]_{\mathcal{A}}) > 0$ for all i . To start, the gambler bets on the set $[\sigma_0]_{\mathcal{A}}$ with a wager such that if he wins, his capital is 2. If he wins, he is done. If he loses, then he bets on the set $[\sigma_1]_{\mathcal{A}}$, and so on. Since the set $\{\sigma_i\}$ may be finite, the gambler may not have a set to bet on at certain stages. This is not an issue, since he may just bet on the whole space. This is functionally equivalent to not betting at all since he wins no money.

The only difficulty now is showing that his capital remains nonnegative. Merkle et al. leave this an exercise for the reader; I give an intuitive argument. It is well-known in probability theory that in a betting strategy one can combine bets for the same effect. (Formally, this is the martingale stopping theorem—see [40].) Hence instead of separately betting on $[\sigma_0]_{\mathcal{A}}, \dots, [\sigma_k]_{\mathcal{A}}$ the gambler will have the same capital as if he just bet on the union $[\sigma_0]_{\mathcal{A}} \cup \dots \cup [\sigma_k]_{\mathcal{A}}$. In the later case, the proper wager would be.

$$\frac{\mu([\sigma_0]_{\mathcal{A}} \cup \dots \cup [\sigma_k]_{\mathcal{A}})}{\mu(\mathcal{X} \setminus ([\sigma_0]_{\mathcal{A}} \cup \dots \cup [\sigma_k]_{\mathcal{A}}))} \leq 1,$$

The inequality follows from

$$\mu([\sigma_0]_{\mathcal{A}} \cup \dots \cup [\sigma_k]_{\mathcal{A}}) \leq 1/2 \leq \mu(\mathcal{X} \setminus ([\sigma_0]_{\mathcal{A}} \cup \dots \cup [\sigma_k]_{\mathcal{A}})).$$

Hence the gambler never wagers (and so loses) more than his starting capital of 1.

Now, assume $z \in \mathcal{X}$ is not Martin-Löf random. Let (U_k) be a Martin-Löf test which covers z . We may assume (U_n) is decreasing. The betting strategy will be as follows which bets on some $x \in \mathcal{X}$. Since $\mu(U_1) < 1/2$ we can start with the computable betting strategy above which will reach a capital of 2 if $x \in U_1$. (Recall, we are not actually betting on U_1 , but the a.e. equal set $\bigcup_i [\sigma_i]_{\mathcal{A}}$. This is not an issue, since the difference is a null Σ_2^0 set. If x is in the difference, then x is not computably random, and so not betting random.)

Now, if the capital of 2 is never reached then $x \notin U_1$ and x is random. However, if the capital of 2 is reached (in a finite number of steps) then we know that $x \in [\sigma]_{\mathcal{A}}$ for some $\sigma = \sigma_i$ (and no other). Further, by the assumptions in the above construction, $\mu([\sigma]_{\mathcal{A}}) > 2^{-k}$ for some k . Then we can repeat the first step, but now we bet that $x \in U_{k+1}$ and attempt to double our capital to 4. Since $\mu(U_{k+1} \mid [\sigma]_{\mathcal{A}}) \leq 1/2$, the capital will remain positive.

Continuing this strategy for capitals of 8, 16, 32, \dots we have a computable betting strategy. If this strategy succeeds on x , then $x \in U_k$ for infinitely many k . Hence x is covered by (U_k) and is not Martin-Löf random. \square

³This follows from Kolmogorov's inequality (proved by Ville, see [14, Theorem 6.3.3 and Lemma 6.3.15 (ii)]) which is a straight-forward application of Doob's submartingale inequality (see for example [40, Section 16.4]).

Remark 9.3. Since there is a universal Martin-Löf test (U_k) , there is a universal computable betting strategy. (The null Σ_2^0 set of exceptions can be handled by being more careful. Choose \mathcal{A} to be basis for the topology, and combine the null cells $[\sigma]_{\mathcal{A}}$ with non-null cells $[\tau]_{\mathcal{A}}$.) However, note that this universal strategy is very different from that of Kolmogorov-Loveland randomness. This is the motivation for the next section.

It is also possible to characterize Martin-Löf randomness by computable randomness. First I give a more formal definition of computable betting strategy.

Remark 9.4. Represent a computable betting strategy as follows. There is a computably indexed family of a.e. decidable sets $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$. These represent the sets being bet on after the wins/loses characterized by $\sigma \in 2^{<\omega}$. From this, we have a computably indexed family $\{B_\sigma\}_{\sigma \in 2^{<\omega}}$ defined recursively by $B_\varepsilon = \mathcal{X}$, $B_{\sigma 1} = B_\sigma \cap A_\sigma$ and $B_{\sigma 0} = B_\sigma \cap (\mathcal{X} \setminus A_\sigma)$. This represents the known information after the wins/loses characterized by $\sigma \in 2^{<\omega}$. It is easy to see that $B_{\sigma 0} \cap B_{\sigma 1} = \emptyset$ and $B_{\sigma 0} \cup B_{\sigma 1} = B_\sigma$ a.e. Then a computable betting strategy can be represented as a partial computable martingale $M: 2^{<\omega} \rightarrow [0, \infty)$ such that

$$M(\sigma 0)\mu(B_{\sigma 0}) + M(\sigma 1)\mu(B_{\sigma 1}) = M(\sigma)\mu(B_\sigma)$$

and $M(\sigma)$ is defined if and only if $\mu(B_{\sigma 0}) > 0$. Again, $M(\sigma)$ represents the capital after a state of σ wins/losses. Say the strategy SUCCEEDS on x if there is some strictly-increasing chain $\sigma_0 \prec \sigma_1 \prec \sigma_2 \prec \dots$ from $2^{<\omega}$ such that $\limsup_{n \rightarrow \infty} M(\sigma_n) = \infty$ and $x \in B_{\sigma_n}$ for all n . Then $x \in \mathcal{X}$ is betting random if there does not exist some $\{A_\sigma\}_{\sigma \in 2^{<\omega}}$ and M as above which succeed on x .

Lemma 9.5. *Fix a computable probability space (\mathcal{X}, μ) . For each computable betting strategy there is a computable probability measure ν on 2^ω , a morphism $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$, and a computable martingale M on $(2^\omega, \nu)$ such that if this betting strategy succeeds on x , then the martingale M succeeds on $T(x)$. Hence $T(x)$ is not computably random on $(2^\omega, \nu)$.*

Proof. Fix a computable betting strategy. Let $M: 2^{<\omega} \rightarrow [0, \infty)$ and $\{B_\sigma\}_{\sigma \in 2^{<\omega}}$ be the as in Remark 9.4. Then define $(2^\omega, \nu)$ by $\nu(\sigma) = \mu(B_\sigma)$. Also, let $T(x)$ map x to the $y \in 2^\omega$ such that $x \in B_{y \upharpoonright n}$ for all n . Then T is a morphism, M also represents a martingale on $(2^\omega, \nu)$, and if the betting strategy succeeds on x then M succeeds on $T(x)$. \square

We now have the following characterizations of Martin-Löf randomness.

Corollary 9.6. *For a computable probability space (\mathcal{X}, μ) , the following are equivalent for $x \in \mathcal{X}$.*

- (1) x is Martin-Löf random.
- (2) No computable betting strategy succeeds on x (i.e. x is betting random).
- (3) For all isomorphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$, $T(x)$ is “Kolmogorov-Loveland random” on $(2^\omega, \nu)$ (i.e. the randomness from Section 9.1).
- (4) For all morphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \nu)$, $T(x)$ is computably random on $(2^\omega, \nu)$.

Proof. The equivalence of (1) and (2) is Theorem 9.2. (1) implies both (3) and (4) since morphisms preserve Martin-Löf randomness (Proposition 7.5).

(4) implies (2): Use Lemma 9.5. Assume x is not betting random. Then there is some morphism T such that $T(x)$ is not computable random.

(3) implies (2): Recall that the definition of betting randomness came from applying the method of Section 8 to Kolmogorov-Loveland randomness. By method (3) of Step 2 in Section 8, x is betting random if and only if (3) holds. (An alternate proof would be to modify Lemma 9.5.) \square

Corollary 9.7. *Computable randomness is not preserved under morphisms. (See comments after Proposition 7.5.)*

Proof. It is well-known that there is an $x \in 2^\omega$ which is computably random on $(2^\omega, \lambda)$ but not Martin-Löf random (see [14, 32]). Then by Corollary 9.6, there is some morphism T such that $T(x)$ is not computably random. \square

Corollary 9.7 was also proved by Bienvenu and Porter [7].

9.3. Step 3: Is the new definition consistent with the former? To show that Martin-Löf randomness equals Kolmogorov-Loveland randomness, we would need to show that “Kolmogorov-Loveland randomness” for all computable probability measures on Cantor space (as in Section 9.1) is preserved by isomorphisms. However, it is not even known if Kolmogorov-Loveland randomness on $(2^\omega, \lambda)$ is base invariant (see Examples 5.9 and 7.11), so I leave this as an open question.

Question 9.8. *Is Kolmogorov-Loveland randomness, as in Section 9.1, preserved under isomorphisms?*

10. ENDOMORPHISM RANDOMNESS

The generalization of Kolmogorov-Loveland randomness given in the last section was, in some respects, not very satisfying. In particular, the definition of Kolmogorov-Loveland randomness on $(2^\omega, \lambda)$ assumes each event being bet on is independent of all the previous events, and further has conditional probability $1/2$. Therefore, at the “end” of the gambling session, regardless of how much the gambler has won or lost, he knows what x is up to a measure-zero set (where x is the string being bet on). This is in contrast to the universal betting strategy given in the proof of Theorem 9.2 (see Remark 9.3), which only narrows x down to a positive measure set when x is Martin-Löf random.

In this section, I now give a new type of randomness which behaves more like Kolmogorov-Loveland randomness. This randomness notion can be defined using both morphisms and betting strategies.

Definition 10.1. Let (\mathcal{X}, μ) be a computable probability space. An **ENDOMORPHISM** on (\mathcal{X}, μ) is a morphism from (\mathcal{X}, μ) to itself. Say $x \in \mathcal{X}$ is **ENDOMORPHISM RANDOM** if for all endomorphisms $T: (\mathcal{X}, \mu) \rightarrow (\mathcal{X}, \mu)$, we have that $T(x)$ is computably random.

Notice the above definition is the same as that given in Corollary 9.6 (4), except that “morphism” is replaced with “endomorphism”.

If the space is atomless, we have an alternate characterization.

Proposition 10.2. *Let (\mathcal{X}, μ) be a computable probability space with no atoms. Then $x \in \mathcal{X}$ is endomorphism random if and only if for all morphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$, $T(x)$ is computably random.*

Proof. Use that there is an isomorphism from (\mathcal{X}, μ) to $(2^\omega, \lambda)$ (Theorem 7.15) and that isomorphisms preserve computable randomness (Theorem 7.9). \square

Also, we can define endomorphism randomness using computable betting strategies as in the previous section.

Definition 10.3. Let (\mathcal{X}, μ) be an atomless computable probability space. Consider a computable betting strategy \mathcal{B} . Let $\{A_\sigma\}_{\sigma \in 2^{<\omega}}, \{B_\sigma\}_{\sigma \in 2^{<\omega}}$ be as in Remark 9.4. Call the betting strategy \mathcal{B} **BALANCED** if it only bets on events with conditional probability $\frac{1}{2}$, conditioned on B_σ (the information known by the gambler at after the wins/loses given by σ). In other words, $\mu(A_\sigma \mid B_\sigma) = 1/2$. Call the betting strategy \mathcal{B} **EXHAUSTIVE** if $\mu(B_{\sigma_n}) \rightarrow 0$ for any strictly increasing chain $\sigma_0 \prec \sigma_1 \prec \dots$. In other words the measure of the information known about x approaches 0.

Theorem 10.4. Let (\mathcal{X}, μ) be an atomless computable probability space and $x \in \mathcal{X}$. The following are equivalent.

- (1) x is endomorphism random.
- (2) There does not exist a balanced computable betting strategy which succeeds on x .
- (3) There does not exist an exhaustive computable betting strategy which succeeds on x .

Proof. (3) implies (2) since balanced betting strategies are exhaustive. For (2) implies (1), assume x is not endomorphism random. Then there is some morphism $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$ such that $T(x)$ is not computably random. Hence there is a computable martingale M which succeeds on $T(x)$. We can also assume this martingale is rational valued, so it is clear what bit is being bet on. This martingale on $(2^\omega, \lambda)$ can be pulled back to a computable betting strategy on (\mathcal{X}, μ) (use the proof of Lemma 9.5, except in reverse). This betting strategy is balanced since M is a balanced “dyadic” martingale.

For (1) implies (3), assume there is some computable, exhaustive betting strategy which succeeds on x . Then from this strategy we can construct a morphism $S: (\mathcal{X}, \mu) \rightarrow ([0, 1], \lambda)$ recursively as follows. Each B_σ will be mapped to an open interval (a, b) of length $\mu(B_\sigma)$. First, map $S(B_\varepsilon) = (0, 1)$. For the recursion step, assume $S(B_\sigma) = (a, b)$ of length $\mu(B_\sigma)$. Set $S(B_{\sigma 0}) = (a, a + \mu(B_{\sigma 0}))$ and $S(B_{\sigma 1}) = (a + \mu(B_{\sigma 0}), b)$. This function S is well-defined and computable since the betting strategy is exhaustive. Also, S is clearly measure-preserving, so it is a morphism. Then using the usual isomorphism from $([0, 1], \lambda)$ to $(2^\omega, \lambda)$, we can assume S is a morphism to $(2^\omega, \lambda)$. Moreover, the set of images $S(B_\sigma)$ describes a cell decomposition \mathcal{A} of $(2^\omega, \lambda)$, and the betting strategy can be pushed forward to give a martingale on $(2^\omega, \lambda)$ with respect to \mathcal{A} (similar to the proof of Lemma 9.5). \square

Now we can relate endomorphism randomness to Kolmogorov-Loveland randomness.

Corollary 10.5. On $(2^\omega, \lambda)$, endomorphism randoms are Kolmogorov-Loveland randoms.

Proof. Every nonmonotonic, computable betting strategy on bits is a balanced betting strategy. Hence every Kolmogorov-Loveland random is endomorphism random. \square

Corollary 10.6. Let (\mathcal{X}, μ) be a computable probability space with no atoms. Then $x \in \mathcal{X}$ is endomorphism random if and only if for all morphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$, $T(x)$ is Kolmogorov-Loveland random.

Proof. If x is endomorphism random on $(2^\omega, \lambda)$, then so is $T(x)$. By Corollary 10.5, $T(x)$ is Kolmogorov-Loveland random. If $T(x)$ is Kolmogorov-Loveland random for all morphisms $T: (\mathcal{X}, \mu) \rightarrow (2^\omega, \lambda)$, then $T(x)$ is computably random for all such T . Therefore, x is endomorphism random. \square

Corollary 10.7. *Computable randomness is not preserved by endomorphisms.*

Proof. It is well-known that there exists a computable random $x \in (2^\omega, \lambda)$ which is not Kolmogorov-Loveland random (see [14, 32]). Then x is not endomorphism random, and is not preserved by some endomorphism. \square

Also, clearly each betting random (i.e. each Martin-Löf random) is an endomorphism random.

I will add one more randomness notion. Say $x \in 2^\omega$ is AUTOMORPHISM RANDOM (on $(2^\omega, \lambda)$) if for all automorphisms $T: (2^\omega, \lambda) \rightarrow (2^\omega, \lambda)$, $T(x)$ is Kolmogorov-Loveland random. It is clear that on $(2^\omega, \lambda)$ we have.

$$(10.1) \quad \text{Martin-Löf} \rightarrow \text{Endomorphism} \\ \rightarrow \text{Automorphism} \rightarrow \text{Kolmogorov-Loveland}$$

We now have a more refined version of Question 9.1.

Question 10.8. *Do any of the implications in formula (10.1) reverse?*⁴

11. FURTHER DIRECTIONS

Throughout this paper I was working with a.e. computable objects: a.e. decidable sets, a.e. decidable cell decompositions, a.e. computable morphisms, and Kurtz randomness—which as I showed, can be defined by a.e. computability. Recall a.e. decidable sets are only sets of μ -continuity, and a.e. computable morphisms are only a.e. continuous maps.

The “next level” is to consider the computable Polish spaces of measurable sets and measurable maps. The a.e. decidable sets and a.e. computable maps are dense in these spaces. Hence, in the definitions, one may replace a.e. decidable sets, a.e. decidable cell decompositions, a.e. computable morphisms, and Kurtz randomness with effectively measurable sets, decompositions into effectively measurable cells, effectively measurable measure-preserving maps, and Schnorr randomness. (This is closely related to the work of Pathak, Simpson and Rojas [33]; Miyabe [31]; Hoyrup and Rojas [personal communication]; and the author on “Schnorr layerwise-computability” and convergence for Schnorr randomness.) Indeed, the results of this paper remain true, even with those changes. However, some proofs change and I will give the results in a later paper.

An even more general extension would be to ignore the metric space structure all together. Any standard probability space can be described uniquely by the measures of an intersection-closed class of sets, or a π -system, which generates the sigma-algebra of the measure. From this, one can obtain a cell decomposition. In the case of a computable probability space (Definition 3.2), each a.e. decidable

⁴Recently, and independently of my work, Tomislav Petrovic has claimed that there are two balanced betting strategies on $(2^\omega, \lambda)$ such that if a real x is not Martin-Löf random, then at least one of the two strategies succeeds on x . In particular, Petrovic’s result, which is in preparation, would imply that endomorphism randomness equals Martin-Löf randomness. Further, via the proof of Theorem 10.4, this result would extend to every atomless computable probability space.

generator closed under intersections is a π -system. The definition of computable randomness on such a general space would be the analog of the definition in this paper.

In particular, this would allow one to define computable randomness on effective topological spaces with measure [21]. In this case the π -system is the topological basis. This also allows one to define Schnorr, Martin-Löf, and weak-2 randomness as well, namely replace, say, Σ_1^0 sets in the definition with effective unions of sets in the π -system. This agrees with most definitions of, say, Martin-Löf randomness in the literature.⁵

Using π -systems also allows one to define “abstract” measure spaces without points. The computable randoms then become “abstract points” given by generic ultrafilters on the Boolean algebra of measurable sets *a la* Solovay forcing.

Another possible generalization is to non-computable probability spaces (on computable Polish spaces). This has been done by Levin [25] and extended by others (see [18, 4]) for Martin-Löf randomness in a natural way using UNIFORM TESTS which are total computable functions from measures to tests. Possibly a similar approach would work for computable randomness. For example, on 2^ω , a uniform test for computable randomness would be a total computable map $\mu \mapsto \nu$ where ν is the bounding measure for μ . This map is enough to define a uniform martingale test for each μ given by $\nu(\sigma)/\mu(\sigma)$. (I showed in Section 2 that this martingale is uniformly computable.) Uniform tests for Schnorr and computable randomness have been used by Miyabe [30].

Also, what other applications for a.e. decidable sets are there in effective probability theory? The method of Section 8 basically allows one to treat every computable probability space as the Cantor space. It is already known that the indicator functions of a.e. decidable sets can be used to define L^1 -computable functions [31].

However, when it comes to defining classes of points, the method of Section 8 is specifically for defining *random* points since such a definition must be a subclass of the Kurtz randoms. Under certain circumstances, however, one may be able to use related methods to generalize other definitions. For example, is the following a generalization of K-triviality to arbitrary computable probability spaces? Let $K = K_M$ where M is a universal prefix-free machine. Recall, a string $x \in 2^\omega$ is K-TRIVIAL (on $(2^\omega, \lambda)$) if there is some b such that

$$\forall n \ K(x \upharpoonright n) \leq K(n) + b$$

where $K(n) = K(0^n)$ and 0^n is the string of 0's of length n . Taking a clue from Section 6, call a point $x \in (\mathcal{X}, \mu)$ K-TRIVIAL if there is some cell decomposition \mathcal{A} and some b such that for all n ,

$$K(x \upharpoonright_{\mathcal{A}} n) < K(-\log \mu([x \upharpoonright_{\mathcal{A}} n]_{\mathcal{A}})) + b.$$

(Here we assume $K(\infty) = \infty$.) Does the \mathcal{A} -name or Cauchy-name of x satisfy the other nice degree theoretic properties of K-triviality, such as being low-for- (\mathcal{X}, μ) -random? (Here I say a Turing degree \mathbf{d} is low-for- (\mathcal{X}, μ) -random if when used as an oracle, \mathbf{d} does not change the class of Martin-Löf randoms in (\mathcal{X}, μ) . Say a point $x \in (\mathcal{X}, \mu)$ is low-for- (\mathcal{X}, μ) -random if its Turing degree is.)

⁵There are some authors [2, 21] that define Martin-Löf randomness via open covers, even for non-regular topological spaces. This will not necessarily produce a measure-one set of random points, where as my method will. All these methods agree for spaces with an effective regularity condition.

If it is a robust definition, how does it relate to the definition of Melnikov and Nies [26] generalizing K-triviality to computable Polish spaces (as opposed to probability spaces)? I conjecture that their definition is equivalent to being low-for- (\mathcal{X}, μ) -random on every computable probability measure μ of \mathcal{X} .

Last, isomorphisms and morphisms offer a useful tool to classify randomness notions. One may ask what randomness notions (defined for all computable probability measures on 2^ω) are invariant under morphisms or isomorphisms? By Proposition 7.5, Martin-Löf, Schnorr, and Kurtz randomness are invariant under morphisms. (This can easily be extended to n -randomness, weak n -randomness, and difference randomness. See [14, 32] for definitions.) However, by Proposition 9.6 (4), there is no randomness notion between Martin-Löf randomness and computable randomness that is invariant under morphisms. Is there such a randomness notion between Schnorr randomness and Martin-Löf randomness? Further, by Theorem 7.9 computable randomness is invariant under isomorphisms. André Nies pointed out to me that this is not true of partial computable randomness since it is not invariant under permutations [5]. In general what can be said of full-measure, isomorphism-invariant sets of a computable probability space (\mathcal{X}, μ) ? The notions of randomness connected to computable analysis will most likely be the ones that are invariant under isomorphisms.⁶

11.1. Acknowledgments. I would like to thank Mathieu Hoyrup and Cristóbal Rojas for suggesting to me the use of isomorphisms to define computable randomness outside Cantor space. I would also like to thank Jeremy Avigad, Laurent Bienvenu, Peter Gács, and Mathieu Hoyrup for helpful comments on a draft of this paper, as well as Joseph Miller and André Nies for helpful comments on this research.

REFERENCES

- [1] Jeremy Avigad. Uniform distribution and algorithmic randomness. *Journal of Symbolic Logic*, To appear.
- [2] Logan M. Axon. *Algorithmically random closed sets and probability*. ProQuest LLC, Ann Arbor, MI, 2010. Thesis (Ph.D.)—University of Notre Dame.
- [3] George Barmpalias, Paul Brodhead, Douglas Cenzer, Seyyed Dashti, and Rebecca Weber. Algorithmic randomness of closed sets. *J. Logic Comput.*, 17(6):1041–1062, 2007.
- [4] Laurent Bienvenu, Peter Gács, Mathieu Hoyrup, Cristóbal Rojas, and A. Shen. Algorithmic tests and randomness with respect to a class of measures. *Proceedings of the Steklov Institute of Mathematics*, 274:41–102, 2011.
- [5] Laurent Bienvenu, Rupert Hölzl, Thorsten Kräling, and Wolfgang Merkle. Separations of non-monotonic randomness notions. *J. Logic and Computation*, 2010.
- [6] Laurent Bienvenu and Wolfgang Merkle. Constructive equivalence relations on computable probability measures. *Ann. Pure Appl. Logic*, 160(3):238–254, 2009.
- [7] Laurent Bienvenu and Christopher Porter. Strong reductions in effective randomness. Submitted.
- [8] Errett Bishop and Douglas Bridges. *Constructive analysis*. 279. Springer-Verlag, 1985.
- [9] Volker Bosserhoff. Notions of probabilistic computability on represented spaces. *J. Universal Computer Science*, 14(6):956–995, 2008.
- [10] Vasco Brattka. Computable versions of Baire’s category theorem. In *Mathematical foundations of computer science, 2001 (Mariánské Lázně)*, volume 2136 of *Lecture Notes in Comput. Sci.*, pages 224–235. Springer, Berlin, 2001.

⁶There is at least one exception to this rule. Avigad [1] discovered that the randomness notion, called UD randomness, characterized by a theorem of Weyl is incomparable with Kurtz randomness; and therefore, it is not even preserved by automorphisms.

- [11] Vasco Brattka, Peter Hertling, and Klaus Weihrauch. A tutorial on computable analysis. In *New computational paradigms*, pages 425–491. Springer, New York, 2008.
- [12] Vasco Brattka, Joseph S. Miller, and André Nies. Randomness and differentiability. *Submitted*.
- [13] Rodney G Downey, Evan Griffiths, and Geoffrey LaForte. On Schnorr and computable randomness, martingales, and machines. *Math. Logic Quart.*, 50(6):613–627, 2004.
- [14] Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic randomness and complexity*. Theory and Applications of Computability. Springer, New York, 2010.
- [15] Willem L. Fouché. The descriptive complexity of Brownian motion. *Adv. Math.*, 155(2):317–343, 2000.
- [16] Cameron E. Freer and Daniel M. Roy. Computable de Finetti measures. *Annals of Pure and Applied Logic*, 2011.
- [17] Péter Gács. Exact expressions for some randomness tests. *Z. Math. Logik Grundlag. Math.*, 26(5):385–394, 1980.
- [18] Péter Gács. Uniform test of algorithmic randomness over a general space. *Theoret. Comput. Sci.*, 341(1-3):91–137, 2005.
- [19] Péter Gács, Mathieu Hoyrup, and Cristóbal Rojas. Randomness on computable probability spaces—a dynamical point of view. *Theory Comput. Syst.*, 48(3):465–485, 2011.
- [20] Peter Hertling and Yongge Wang. Invariance properties of random sequences. *J.UCS*, 3(11):1241–1249 (electronic), 1997.
- [21] Peter Hertling and Klaus Weihrauch. Random elements in effective topological spaces with measure. *Inform. and Comput.*, 181(1):32–56, 2003.
- [22] John M. Hitchcock and Jack H. Lutz. Why computational complexity requires stricter martingales. *Theory Comput. Syst.*, 39(2):277–296, 2006.
- [23] Mathieu Hoyrup and Cristóbal Rojas. Computability of probability measures and Martin-Löf randomness over metric spaces. *Inform. and Comput.*, 207(7):830–847, 2009.
- [24] Bart Kastermans and Steffen Lemp. Comparing notions of randomness. *Theoret. Comput. Sci.*, 411(3):602–616, 2010.
- [25] L. A. Levin. Uniform tests for randomness. *Dokl. Akad. Nauk SSSR*, 227(1):33–35, 1976.
- [26] Alexander Melnikov and André Nies. K-triviality in computable metric spaces. *Proceedings of the American Mathematical Society*, To appear.
- [27] Wolfgang Merkle. The Kolmogorov-Loveland stochastic sequences are not closed under selecting subsequences. *J. Symbolic Logic*, 68(4):1362–1376, 2003.
- [28] Wolfgang Merkle, Nenad Mihailović, and Theodore A. Slaman. Some results on effective randomness. *Theory Comput. Syst.*, 39(5):707–721, 2006.
- [29] Wolfgang Merkle, Joseph S. Miller, André Nies, Jan Reimann, and Frank Stephan. Kolmogorov-Loveland randomness and stochasticity. *Ann. Pure Appl. Logic*, 138(1-3):183–210, 2006.
- [30] Kenshi Miyabe. Truth-table Schnorr randomness and truth-table reducible randomness. *MLQ Math. Log. Q.*, 57(3):323–338, 2011.
- [31] Kenshi Miyabe. l^1 -computability, layerwise computability and Solovay reducibility. *Submitted*.
- [32] André Nies. *Computability and randomness*, volume 51 of *Oxford Logic Guides*. Oxford University Press, Oxford, 2009.
- [33] Noopur Pathak, Cristóbal Rojas, and Stephen G. Simpson. Schnorr randomness and the Lebesgue differentiation theorem. *Proceedings of the American Mathematical Society*, To appear.
- [34] Marian B. Pour-El and J. Ian Richards. *Computability in analysis and physics*. Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1989.
- [35] Claus-Peter Schnorr. A unified approach to the definition of random sequences. *Math. Systems Theory*, 5:246–258, 1971.
- [36] Matthias Schröder. Admissible representations for probability measures. *MLQ Math. Log. Q.*, 53(4-5):431–445, 2007.
- [37] Javier Gonzalo Silveira. Invariancia por cambio de base de la aleatoriedad computable y la aleatoriedad con recursos acotados. Master’s thesis, University of Buenos Aires, 2011.
- [38] V. V. V’yugin. Ergodic theorems for individual random sequences. *Theoret. Comput. Sci.*, 207(2):343–361, 1998.
- [39] Klaus Weihrauch. *Computable analysis: An introduction*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2000.

- [40] David Williams. *Probability with martingales*. Cambridge Mathematical Textbooks. Cambridge University Press, Cambridge, 1991.